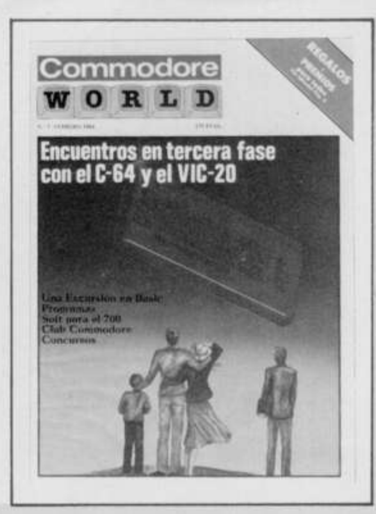
Commodore WORLD

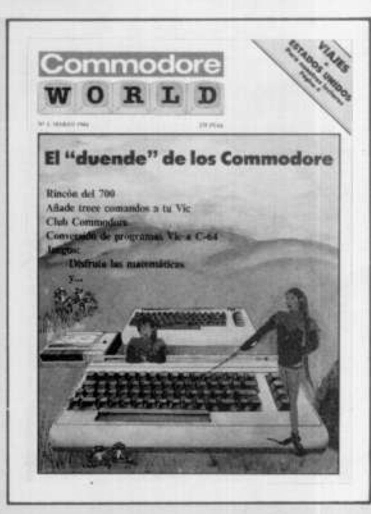
Nº 4. MAYO 1984

275 PTAS.

VIAJES CONCURSOS ....DINERO....











"Commodore World" aparte de venderse generalmente en kioscos, se encuentra asimismo a la venta en los siguientes distribuidores de Commodore y librerías.

ALICANTE

Casa Wagner, c/ Juan Carlos I, 37. Elda.
Tel. (965) 39 03 96.
Ferretería Progreso. c/ General Jordana, 28. Elda.
Tel. (965) 38 11 45.

BADAJOZ

Control Sistemas. Avda. Santa Marina, 25A.
Tel. (924) 25 88 00.

BARCELONA

Computer Service. Avda. Abad Marcet, 325. Tarrasa.
Tel. (93) 788 63 77
Librería Emilia Pérez Radua. c/Mayor, 35. Castellar del Vallés. Tel. (93) 714 89 51.
Librería Michel. Ronda Guinardo, 1. Sardañola.
Mozart. c/ Jaime I, 145, Mollet. Tel. (93) 593 75 01

BURGOS

E.I.S.A. c/ Madrid, 4. Tel. (947) 20 46 24.

CANARIAS

Relax. Rambla de Pulido, 85. Santa Cruz de Tenerife.
Tel. (922) 28 37 05.

CIUDAD REAL

CISA. c/ D. Víctor, 15. Tomelloso.
Tels. (926) 51 00 07 - 51 15 72.
Electrónica Turrillo. c/ Pedrera Baja, 7.
Tel. (926) 22 38 67.

LA CORUÑA

Cetronic, S. L. c/ Palomar, 22 Bajo. Tel. (981) 27 26 54.
Photo Copy. c/ Teresa Herrera, 9. Tel. (981) 21 34 21.
Sanlusa, S. L. San Luis, 46 al 50. Tel. (981) 23 07 49.

GERONA

Digit Informática. c/ Banyoles, 13. Barri de Sant Roc.
Olot. Tel. (972) 26 94 01.

HUELVA

Computerlog, S. A. c/ Tendaleras, 15.
Tel. (955) 25 81 99

MADRID

Chips + Tips - C/Puerto Rico, 21-23
Librería Díaz de Santos. c/ Lagasca, 98.
Librería García Pena. c/ Cavanilles, 52.
Librería Ra-Ma. c/ Chiquinquira, 28. Urb. Columbia.
Microtronic. c/ Quinta del Sol, 39. Las Rozas.
Multiprensa. c/ Castelló, 37.
País de los Microordenadores. Bravo Murillo, 18.
Sonitel Central. c/ Clara del Rey, 24.

MURCIA

Procoinsa. Ronda Norte, 27. Tel. (968) 23 94 49.

PALMA DE MALLORCA

AEF. c/ Balmes, 13. Tel. (971) 27 36 66.

PONFERRADA

MicroBierzo. c/ Carlos I, 2. Tel. (987) 41 74 21.

SALAMANCA

Compusoft. c/ Arco, 1.

SANTANDER

Librería Hernández. San Francisco, 15. Tel. (942) 22 53 30

SANTIAGO DE COMPOSTELA

GestyComputer. Avda. Romero Donallo, 25.
Tel. (981) 59 87 54.

SEVILLA

Papelería Mora. c/ Santa Cruz, 5.

TARRAGONA

Oficomplet. Plaza de la Cinta, 6. Tortosa.
Tel. (977) 44 14 50.

ZARAGOZA

ADA Computer. Centro Independencia.
Paseo Independencia, 24-26. Tel. (976) 29 85 62.

TIRADA DE COMMODORE WORLD
Certificación de 28.000 ejemplares

Según certificado firmado por la auditoría internacional DELOITTE HASKINS SELLS, S. A.
el 5 de marzo de 1984,
la tirada de Commodore World es de 28.000 ejemplares.
Este certificado se encuentra en las oficinas de S.I.M.S.A. para comprobación
de cualquier persona o entidad que lo desee.



SUMARIO 4

Mayo 1984

CONTENIDO — PAG.

PROXIMO NUMERO

- Ahorrando Memoria con el VIC
- Programa de Evaluación para Colegios
- Programa de Gestión de Ficheros 8000 y el 700
- La Nueva ROM de Kernal del C-64
- Entiende mejor tu ordenador con un divertido juego
- Juegos
- Club Commodore
- Y todas vuestras colaboraciones

¡OS ESPERAMOS!

Commodore World es redactado en colaboración entre Microelectrónica y Control-Commodore y SIMSA

EQUIPO

José Luis ERRAZQUIN; Miguel A. H. HIDALGO; María LOPEZ; Juan MARTINEZ; Pere MASATS; Victoria MORALES; Fernando M. RODRIGUEZ; Diego ROMERO; Albert SANGLAS; Jordi SASTRE; Valerie SHANKS...
...Y NUESTROS LECTORES

SIMSA
Coordinador María López
Pedro Murugura, 4-8ºB — Madrid-16 — Tel.: (91) 259 54 78
Delegación en Barcelona:
Sant Gervasi de Cassoles, 39 despacho 4
Barcelona-22 — Tel.: (93) 212 73 45

MICROELECTRONICA Y CONTROL-COMMODORE
Coordinador Pere Masats
Taquígrafo Serra, 7-5º
Barcelona-29
Tels.: (93) 250 51 03/02

Imprime: IBERDOS, S. A. Germán Pérez Carrasco, 24. Madrid-27
Depósito Legal: M-2944-1984

EDITORIAL

SUPER-INTERESANTISIMO

Y vamos de concursos y premios

1 Viajes

2 Dinero y Video-Juegos

3 Gran Concurso Escolar de Programación

1 VIAJES

Hace un mes —20 de marzo— decíamos que se acababa de recibir la suscripción nº 4154. Hoy —18 de abril— a las 7 de la tarde acabamos de poner número a la 4.701. Sin necesidad de ser un genio matemático, se deduce rápidamente que se han recibido 547 suscripciones en un mes y que faltan solamente para llegar al número 5.000 (para nuestros nuevos lectores aclararemos que el suscriptor que haga el número 5.000 ganará automáticamente un viaje a Estados Unidos y se sorteará otro a Inglaterra entre los 4.999 primeros) lo que quiere decir que es muy posible que en nuestro próximo número de junio podamos comunicar el nombre del suscriptor 5.000, y por tanto ganador del viaje a EE.UU. y anunciar la fecha, hora y lugar en que tendrá lugar el sorteo para Inglaterra.

2 DINERO Y VIDEO-JUEGOS

Como ya anunciamos desde nuestro primer número, en junio y en diciembre tendrá lugar un sorteo entre los lectores que nos envían trucos y programas para su publicación. Nuestra primera idea, como sabéis, fue hacerlo por grupos de edades, pero en vista de que la mayoría no nos envían la edad se procederá a sorteo general. Habrá premios de 15.000, 10.000 y 5.000 pesetas así como programas de juegos y relojes-calculadoras para los programadores y de 3.000, 2.000 y 1.000 y programas de juegos para Magia. El sorteo será público y en nuestro próximo número anunciaremos lugar, día y hora. Recordad que es preferible que enviéis los programas en cinta a ser posible. Si vienen en listado pueden retrasarse en su publicación, y que devolvemos las cintas con un programa de regalo.

3 GRAN CONCURSO ESCOLAR. ATENCION COLEGIOS, ESCOLARES Y EGB, BUP Y FP

Uno de los grandes intereses del equipo de Commodore World es la Informática en la Educación y la Educación en la Informática. Estamos convencidos de que en muy pocos años el ordenador va a ser un utensilio tan familiar y utilizado, tanto en casa como en el trabajo, como lo es hoy un teléfono una máquina de escribir o una fotocopiadora. España adolece hoy de una falta inmensa de programas educativos. Pues bien. Commodore World anuncia hoy la convocatoria del 1.er Concurso de Programas y Juegos Educativos para Escolares y Colegios en los niveles de EGB, BUP y FP. Habrá importantes premios tanto para los escolares individuales que hayan confeccionado los programas ganadores como para los centros escolares que representen. El fallo del concurso tendrá lugar en el mes de diciembre y las bases del mismo serán anunciadas en nuestro próximo número de junio. Así que colegios y escolares, ya podéis poneros a trabajar. Todos los trabajos presentados deben ser originales e inéditos.

SERVICIO DE CINTAS (¡OTRA VEZ!)

Aunque parezcamos unos pesados, queremos recordar a todos que el SERVICIO DE CINTAS incluye cualquier programa aparecido en la revista **EXCEPTO las colaboraciones de lectores**. Su nombre y dirección aparece siempre junto al programa para que podáis poneros directamente en contacto unos con otros.

...FALTAN EN ESTE NUMERO...

El mes pasado anunciamos para este número un Programa de Evaluación para Colegios. Hemos tenido que aplazar un mes su publicación debido a que al traducirlo y adaptarlo al sistema educativo español hemos tenido que ir efectuando diversas comprobaciones y a la hora de entrar en imprenta no habíamos podido efectuar la comprobación final y no quisimos arriesgarnos a que se "colara" un error que podría ser de importancia.

Asimismo, el artículo anunciado "Ahorrando Memoria en el VIC", por una razón tan sencillamente tonta como el no habernos llegado a tiempo en el Correo, aunque se nos han enviado 2 veces... lo que nos lleva a otro tema...

...SERVICIO DE CORREOS...

A la prensa se le llama el cuarto poder y aunque no es la misión de esta revista ejercer estos "derechos de poderío" y la denuncia de servicios o instituciones, por lo que nos afecta el tema, metemos en "La Cárcel de Papel" (Como decía la entrañable y desaparecida "La Codorniz") al Servicio de Correos condenado con todos los agravantes, que lleva más de un mes volviéndonos locos a todos, suscriptores, redacción y diversos otros amigos. Nuestro número 3 se entregó en Correos el día 2 de abril. Hay un importante número de suscriptores que no lo han recibido hasta el 15 ó 16 del mes y otros que hoy día 18 todavía están esperando (Bueno, no esperan porque se les ha enviado 2.ª copia, pero no se trata de eso) y un anunciante de Barcelona a quien se le envió el día 5 nos acaba de llamar diciendo que lo acaba de recibir y con matasellos... ¡del día 10!... Todo un récord... se envía el 5, se matasella el 10, se distribuye el 18. ¿Quién da más en lo que a velocidad eficiencia y seriedad se refiere? Por favor, señores de Correos, vivimos a las puertas del siglo XXI y en el siglo XVI, en silla de posta, el correo tardaba solamente una semana de Madrid a Barcelona.

MICROELECTRONICA Y CONTROL Y SIMSA (COMMODORE WORLD)

Son dos compañías totalmente independientes una de la otra. Su único interés en común es el servicio a los usuarios de Commodore. Commodore World publica dentro de sus páginas lo correspondiente a la antigua revista Club Commodore que publicaba Microelectrónica; hay una colaboración entre las dos firmas en el equipo de redacción de la revista; Commodore World absorbio a los suscriptores de Club Commodore comprometiéndose a seguir cumpliendo esa suscripción hasta su fecha de renovación; las dos firmas mantienen una magnífica y cordial relación y un espíritu de cooperación pero sin ningún otro interés común. Esperamos que esto aclare el confusionismo que algunos lectores parecen haber sufrido.

...Y CUIDADO CON LAS IMITACIONES...

Existe una cierta confusión, entre un número de lectores que nos han llamado consultándonos (¡o increpándonos!) sobre programas aparecidos en alguna revista que **no es nuestra** o quejándose de haber recibido un segundo reembolso, de una segunda publicación, que no habían pedido. Queremos dejar claro que la única revista que publicamos sobre Commodore es **COMMODORE WORLD** y que cualquier otra publicación que utilice un nombre similar no es nuestra ni tiene nada que ver con nosotros. Asimismo, nosotros solamente enviamos cobro sobre suscripciones a aquellas personas que específicamente lo hayan solicitado por boletín, carta o teléfono. Nunca hemos intentado el forzar una suscripción no solicitada.

INFORMAT Y CORTE INGLES

Recordamos a todos que estaremos en Informat y que estaremos en el Corte Inglés.

PRECIO DE COMMODORE WORLD

Lamentamos que a partir del **1º de julio** no tenemos más remedio que subir el precio de nuestra publicación en un 15%. El número de páginas ha ido en aumento gradualmente, todos los meses desde que comenzamos con 42 y asimismo han aumentado los costes de producción y correo. Por consiguiente a partir del 1º de julio, la suscripción por un año será de 2.530 pesetas y la venta la público de 315 pesetas por ejemplar.

VIAJES A ESTADOS UNIDOS E INGLATERRA PARA NUESTROS SUSCRIPTORES

¡CUIDADO!
Solamente la revista COMMODORE WORLD publicada por S.I.M.S.A. está ofreciendo esta oportunidad a sus socios-suscriptores

BOLETIN DE SUSCRIPCION — commodore world

NOMBRE .... EDAD ....
DIRECCION ....
POBLACION .... (.............) PROVINCIA ....
TELEF. .... MARCA Y MODELO DEL ORDENADOR ....
....
CIUDAD DONDE LO COMPRO .... DISTRIBUIDOR ....
APLICACIONES A LAS QUE PIENSA DESTINAR EL EQUIPO ....
....
....

Deseo iniciar la suscripción con el nº ☐
Adjunto cheque de 2.200 pesetas ☐ Firma,
Reembolso más gastos del mismo
al recibir el primer nº de la suscripción ☐
(Enviar a la direccion del dorso)
Extranjero $35. Solamente por correo aéreo

DESEO SUSCRIBIRME A **COMMODORE WORLD** POR UN AÑO AL PRECIO DE 2.200 PTS. DICHA SUSCRIPCION ME DA DERECHO, NO SOLO A RECIBIR LA REVISTA (ONCE NUMEROS ANUALES) SINO A PARTICIPAR EN LAS ACTIVIDADES QUE SE ORGANICEN EN TORNO A ELLA Y QUE PUEDEN SER COORDINACION DE CURSOS DE BASIC, INTERCAMBIOS DE PROGRAMAS, CONCURSOS, ETC.

EJEMPLARES ATRASADOS DE COMMODORE WORLD

Commodore WORLD

1	2	3

Precio por ejemplar 275 ptas.

Forma de pago: sólo por cheque

Peticionario ....
Calle .... **Nº** ....
Población .... **D. P.** .... **Provincia** ....
Incluyo cheque por valor de .... **pesetas + 75 de gastos de envío.**

SERVICIO DE CINTAS

Título del programa .... publicado en nº ☐
Título del programa .... publicado en nº ☐
Título del programa .... publicado en nº ☐
Título del programa .... publicado en nº ☐

Precio por cinta 850 pesetas. Gastos de envío 75 pesetas. Forma de pago: sólo por cheque.

Peticionario ....
Calle .... Nº ....
Población .... D.P. .... Provincia ....
Incluyo cheque sólo por .... **pesetas.**

EJEMPLARES ATRASADOS DE «CLUB COMMODORE»

Para poder satisfacer la creciente demanda de números atrasados de nuestra Revista, agotada en todas sus ediciones, hemos puesto en marcha un Servicio para suministrar fotocopias de los ejemplares que nos sean solicitados. Para recibir las fotocopias de una o de varias ediciones, no hay más que enviarnos el boletin con los datos indicados.

SERVICIO DE FOTOCOPIAS.— NUMERO DE LA EDICION SOLICITADA.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

(Poner una X debajo del número de edición pedido)

Peticionario: D. ....
Calle .... nº ....
Población .... D.P. .... Provincia ....

Forma de pago sólo por cheque **Precio de la edición fotocopiada: 250 ptas.**

La colección completa del 0 al 15: 2.500 ptas. + 100 ptas. por gastos de envío

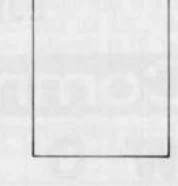
Incluyo cheque por .... **ptas.**

Commodore

W O R L D

Pedro Muguruza, 4-8º B
Teléf.: 259 54 78
MADRID-16

Sant Gervasi de Cassoles, 39-despacho 4
Teléf.: 212 73 45
BARCELONA-22

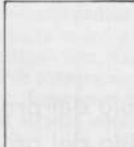


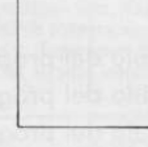
Commodore

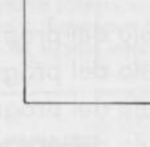
W O R L D

Pedro Muguruza, 4-8º B
Teléf.: 259 54 78
MADRID-16

Sant Gervasi de Cassoles, 39-despacho 4
Teléf.: 212 73 45
BARCELONA-22







Commodore

W O R L D

Pedro Muguruza, 4-8º B
Teléf.: 259 54 78
MADRID-16

Sant Gervasi de Cassoles, 39-despacho 4
Teléf.: 212 73 45
BARCELONA-22



Commodore

W O R L D

Pedro Muguruza, 4-8º B
Teléf.: 259 54 78
MADRID-16

Sant Gervasi de Cassoles, 39-despacho 4
Teléf.: 212 73 45
BARCELONA-22

INDICE DE ANUNCIANTES

Clave para interpretar los listados

Todos los listados que se publican en esta Revista han sido ejecutados en el modelo correspondiente de la gama de ordenadores COMMODORE. Para facilitar la edición de los mismos en la Revista y para mejorar su legibilidad por parte del usuario, se les ha sometido a ciertas modificaciones mediante un programa escrito especialmente para ello. Para los programas destinados a los ordenadores VIC-20 y COMMODORE 64, en los que se usan frecuentemente las posibilidades gráficas del teclado, se han sustituido los símbolos gráficos que aparecen normalmente en los listados por una serie de letras entre corchetes [] que indican la secuencia de teclas que se deben pulsar para obtener el carácter deseado. A continuación se da una tabla para aclarar la interpretación de las indicaciones entre corchetes:

[CRSRD] = Tecla cursor hacia abajo (sin SHIFT)

[CRSRU] = Tecla cursor hacia arriba (con SHIFT)

[CRSRR] = Tecla cursor a la derecha (sin SHIFT)

[CRSRL] = Tecla cursor a la izquierda (con SHIFT)

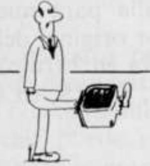
[HOME] = Tecla CLR/HOME (sin SHIFT)

[CLR] = Tecla CLR/HOME (con SHIFT)

Las indicaciones [BLK] a [YEL] corresponden a la pulsación de las teclas de 1 a 8 junto a la tecla CTRL. Lo mismo sucede con [RVSON] y [RVSOF] respecto a la tecla CTRL y las teclas 9 y 10.

El resto de las indicaciones constan de la parte COMM o SHIF seguidas de una letra, número o símbolo —por ejemplo [COMM+] o [SHIFA]—. Esto indica que para obtener el gráfico necesario en el programa deben pulsarse simultáneamente las teclas COMMODORE (la que lleva el logotipo) o una de SHIFT y la tecla indicada por la letra, el número o el símbolo, en el ejemplo anterior: COMMODORE y + o SHIFT y A, respectivamente.

En los signos gráficos además se cuenta el número de veces que aparece. Por ejemplo, [7 CRSRR] equivale a 7 pulsaciones de la tecla cursor a la derecha y [3 SPC] tres pulsaciones de la barra espaciadora. ■



METEDURAS DE PATA

MEJORA DE PROGRAMAS

Para eliminar los problemas de sonido en el programa "Artillería" para el C-64, publicado en nuestro número 2 de marzo, debe efectuarse un cambio en la línea 3520:

```
3520 FOR NC=LC TO VL: POKE
     NC,0:NEXT
```

El programa "Base de Datos" publicado en nuestro número 3 de abril, funciona perfectamente en disco, pero a veces da error en cassette. Para subsanar esta deficiencia se deben efectuar los siguientes cambios:

En el listado del VIC-20 las líneas 37 y 40 deben ser:

```
37 PRINT"[RVSON]D[RVSOFF]ISCO
   [SPC]O[SPC][RVSON]C[RVSOFF]
   INTA?":GOSUB63:IF A$="C"
   THEN40
40 OPEN1,1,1,T$:J=1:GOTO89
```

En el listado del C-64 la línea 400 debe ser:

```
400 OPEN1,1,1,T$:J=1:GOTO890
```

ERRORES DE IMPRENTA

En el artículo "PRINTS SOFISTICADOS" se omitió el nombre del autor: PERE MASATS.

En la sección DE TOMO Y LOMO aparecieron tres libros pero no sus precios, que son:

BASIC BASICO	1.300 pts.
VIC-20 GUIA DEL USUARIO	2.300 pts.
GUIA DEL ORDENADOR PERSONAL PET/CBM	2.250 pts.

EN CORRECCIONES DEL MANUAL los comandos del soporte del DOS del C-64 son:

% NOMBRE = LOAD "NOMBRE",8,1
↑ NOMBRE = LOAD "NOMBRE",8 + RUN
←NOMBRE = SAVE "NOMBRE",8

La línea 530 del programa para leer el directorio debe ser:

```
530 IF A$()CHR$(96) THEN IF A$()CHR$
(160) THEN B$=B$+A$
```

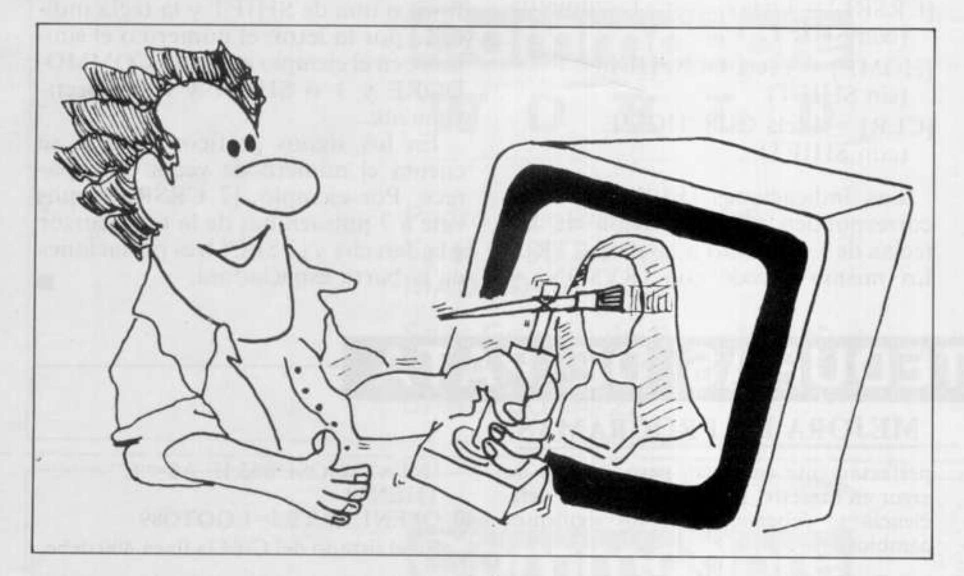
La página 28 del número 3 fue una repetición errónea de la pág. 40 del nº 2, por lo que el texto no se corresponde.

Dibujando con los Commodore

En este programa el cursor se convierte en camaleón, cambiándose de color y carácter con una sola pulsación de tecla, mientras que tú dibujas a tu aire con el "joystick"

El programa **Dibujar** te permite "dibujar" en la pantalla del VIC-20 a pleno color, convirtiendo el carácter del cursor en diversas letras y gráficos y cambiando el color mediante la pulsación de un par de teclas.

El "joystick" se utiliza para desplazar el cursor por la pantalla. El cursor empieza siendo un signo de más de color rojo. Mediante la pulsación de una de las teclas numéricas de 1 a 8 cambia el color de cursor al color de dicha tecla. Si se pulsa otra tecla, el cursor se convierte en el carácter representado por ella, y la pulsación de la tecla "shift" más otra tecla llamará el carácter presentado a la derecha de la parte delantera de la tecla.



También resulta fácil no imprimir nada, es decir, un espacio en blanco en una sección determinada. Para realizar esto, se pulsa la tecla 2 para que el cursor se convierta en el color blanco. El cursor seguirá emitiendo una luz verde intermitente a medida que se desplaza, para que resulte fácil de localizar, pero el carácter que deja atrás será blanco, y, por lo tanto, invisible contra el fondo blanco de la pantalla.

La mayoría de las rutinas de inicialización son parecidas a las del programa "Tiro al Blanco" que presentamos en **Commodore World** nº 1. Para solucionar el problema de la misteriosa memoria móvil del VIC-20 en las diversas configuraciones de aplicación, se almacena la verdadera memoria de pantalla y las posiciones de memoria en color en CHAR y CSCREEN, se calcula la diferencia, DF, y se inicializa la variable E como el fin de la pantalla para nuestros propósitos. El color original del cursor, CO, se inicializa en 2 (rojo), y el carácter del cursor, CURS, en 91, el signo de más.

Variables Adicionales

Sin embargo, se definen varias variables que no figuraban en el artículo "Tiro al Blanco". Dos de estas variables, PA y PB, son las direcciones de memoria de las puertas utilizadas para la lectura de los pulsadores del "joystick". La tercera, DD, es la dirección de lo que se llama el registro de direcciones de datos (DDR).

Los "joysticks" utilizados tienen cinco pulsadores. Uno al norte, uno al sur, otro al este y otro al oeste, más el quinto, el botón de "disparo". El

VIC-20

C-64

```
1 REM VERSION VIC-20
10 REM **********
20 REM *        *
30 REM * DIBUJA *
40 REM *        *
50 REM **********
60 PRINT"[CLR][2CRSRD]"
70 PRINTTAB(7)"[RVSON][RED]DIBUJA[RVSOFF]
[BLU][2CRSRD]"
80 PRINT"*USA[SPC]EL[SPC]JOYSTICK[SPC]
PARA"
85 PRINT"[SPC]DIBUJAR."
90 PRINT"*PULSA[SPC]UN[SPC]NUMERO[SPC]
PARA"
100 PRINT"[SPC]CAMBIAR[SPC]EL[SPC]CO
LOR[SPC]DEL[2SPC]CURSOR."
110 PRINT"*OTRAS[SPC]TECLAS[SPC]CAMB
IAN"
120 PRINT"[SPC]LA[SPC]FORMA[SPC]DEL[SPC]
CURSOR."
130 PRINT"[SPC]PULSA[SPC]EL[SPC]BOTO
N[SPC]DE"
140 PRINT"DISPARO[SPC]PARA[SPC]EMPEZ
AR."
150 PRINT"[2CRSRD]"
160 PRINTTAB(3)"[RVSON][GRN]PULSA[SPC]
UNA[SPC]TECLA[BLU]"
170 GETA$:IFA$=""THEN170
180 CURSR=91
190 CO=2
200 PRINT"[CLR]"
210 CSCREEN=37888+4*(PEEK(36866)AND1
28)
220 CHAR=4*(PEEK(36866)AND128)+64*(P
EEK(36869)AND120)
230 E=CHAR+505
240 B1=CHAR
250 DF=CSCREEN-CHAR
260 DD=37154:PA=37137:PB=37152:POKE3
7139,0
270 GOTO350
280 REM LEE JOYSTICK
290 POKEDD,127:S3=-((PEEK(PB)AND128)
=0):POKEDD,255
300 P=PEEK(PA):FR=-((PAND32)=0)
310 S0=((PAND4)=0)
320 S2=((PAND16)=0)
330 S1=-((PAND8)=0)
340 RETURN
350 POKEB1,CURSR
360 POKEB1+DF,3
370 GOSUB650
380 POKEB1+DF,CO
390 GOSUB290
400 CT=PEEK(197):IFCT=64GOTO470
410 GETA$:IFA$=""GOTO470
420 RE=ASC(A$)
430 IFRE>56ORRE<49GOTO460
440 CO=RE-1
450 GOTO470
460 CURSR=ASC(A$)
470 IFS3<>1GOTO520
480 IFS3<>1GOTO520
490 B1=B1+1
500 IFB1>ETHENB1=E
510 GOTO350
520 IFFR=1THENPRINT"[CLR]"
530 IFS2<>-1GOTO570
540 B1=B1-1
550 IFB1<CHARTHENB1=CHAR
560 GOTO350
570 IFS0<>-1THENGOTO610
580 B1=B1-22
590 IFB1<CHARTHENB1=B1+22
600 GOTO350
610 IFS1<>1GOTO350
620 B1=B1+22
630 IFB1>ETHENB1=B1-22
640 GOTO350
650 POKE36878,15
660 POKE36874,255
670 POKE36874,0
680 RETURN
```

estado de estos pulsadores se lee a través de dos puertas, utilizando diferentes bits para diferentes pulsadores. Cada bit puede ser utilizado solamente para la entrada o la salida en un momento determinado, así que hay que indicarle al ordenador lo que tiene que hacer.

Existe una puerta que no se utiliza para otra cosa a lo largo del juego, así que puede ser inicializada como entrada mediante un simple comando "Poke" en el registro de dirección de datos en 37139. Esto se realiza en la línea 260.

Sin embargo, el otro registro de dirección de datos también se utiliza para la entrada/salida del teclado. Si se inicializara solamente como entrada, sería imposible utilizar el teclado para otras tareas a lo largo del programa. Este registro de dirección de datos se define como DD, se hace un "Poke" para la entrada antes de cada lectura del "joystick" y vuelve a su estado normal en seguida.

A diferencia de otras operaciones de "Peek", o búsqueda de un número determinado dentro de una sola posición de memoria, en este caso los bits individuales dentro de un byte son significantes. Es posible descubrir el estado de un bit determinado mediante el uso del operador AND. Si se realiza una operación AND de un número con la correcta potencia de dos, se puede saber cómo se ha inicializado dicho bit.

La lectura del "joystick" se hace en las líneas 290 a 340. Primero, el DD se inicializa como entrada haciendo un "Poke" en DD con 127. A continuación, el pulsador tres se lee mediante una operación "Peek' en la puerta PB, y una operación AND de 128 con dicho valor. Finalmente, DD se vuelve a su estado normal mediante una operación de "Poke" con 255.

Los otros cuatro pulsadores se leen mediante una operación "Peek" en la

CENTRO DE INFORMATICA
Las Rozas - Majadahonda
EMPEZAMOS
Cursillos en BASIC cada 15 días
Directamente en ordenadores VIC-20, COMMODORE 64 SPECTRAVIDEO
Tfno.: 637 31 51

```
10 REM ***********
20 REM *         *
30 REM * DIBUJA  *
35 REM *   C-64  *
40 REM *         *
50 REM ***********
60 DIMD(10,2)
70 DATA-40,40,0,-1,-41,39,0,1,-39,41

80 FORX=1TO10
90 READD(X,1)
100 NEXTX
110 V=53248
120 PRINT"[CLR]"
130 B1=1024
140 DELTA=1
150 POKE53281,1
160 VOLUME=54296
170 VCE=54273
180 WAVE=54276
190 ATTACK=54277
200 SUS=54278
210 CURSR=43:CO=2
220 GOTO300
230 JV=PEEK(56320)
240 J1=JVAND16
250 F1=15-(JVAND15)
260 JV=PEEK(56321)
270 J2=JVAND16
280 F2=15-(JVAND15)
290 RETURN
300 GOSUB230
310 GETA$:IFA$<>""THENGOSUB430
320 GOSUB480
330 DELTA=0:IFD(F1,1)<>0THENDELTA=D(
F1,1)
335 IFD(F2,1)<>0THENDELTA=D(F2,1)
340 B1=B1+DELTA
350 IFB1>2023THENB1=B1+(D(F1,1)*-1)
360 IFB1<1024THENB1=B1+(D(F1,1)*-1)
370 IFB1>2023THENB1=B1+(D(F2,1)*-1)
380 IFB1<1024THENB1=B1+(D(F2,1)*-1)
390 POKEB1,CURSR:POKEB1+54272,CO
400 IFCO=1THENPOKEB1+54272,3:POKEB1+
54272,CO
410 IFJ1=0ORJ2=0THENPRINT"[CLR]"
420 GOTO300
430 IFA$<"1"ORA$>"8"THEN460
440 CO=VAL(A$)-1
450 RETURN
460 CURSR=ASC(A$)
470 RETURN
480 POKEVOLUME,15
490 POKEWAVE,33
500 POKEATTACK,128
510 POKESUS,128
520 POKEVCE,72
530 POKEVOLUME,0
540 RETURN
```

puerta PA, el almacenamiento de dicho valor en P, y una operación AND de P, con 4, 8, 16 y 32. Esto proporciona el estado del pulsador 0, 1, 2 y el botón de disparo, respectivamente. Todo esto se puede realizar varias veces por segundo, para que el programa tenga siempre disponible el estado actual del "joystick".

Dibujando en pantalla

Se dibuja en pantalla haciendo un "Poke" con el número que corresponde al carácter del cursor en la posición de memoria almacenada en la variable B1, y con el color deseado en B1+DF. La variable B1 se modifica según la dirección en que se encuentra el "joystick".

Al igual que en "Tiro al Blanco", B1 puede ser incrementada o decrementada por uno desplazando el "joystick" a la derecha o a la izquierda. Sin embargo, también es posible desplazarlo hacia arriba. Para hacer esto, B1 es decrementada por 22, o una línea entera. Para desplazarlo hacia abajo, B2 se incrementa por 22. En cualquier caso, el programa no permitirá que B1 sobrepase E (fin de pantalla), o que sea menor que CHAR (el principio de la memoria de pantalla).

La mayoría de las rutinas de iniciación son parecidas a las del programa "Tiro al Blanco". Sin embargo, se definen varias variables que no figuraban en dicho artículo

Aún cuando el ordenador esté verificando constantemente el movimiento del "joystick", también está atento para ver si se ha pulsando cualquier otra tecla. El ordenador sabe si una tecla está siendo pulsada haciendo un "Peek" en la posición 197. Si el valor contenido en 197 es igual a 64, no hay ninguna tecla pulsada. De no ser así, el programa lee A$. Si A$ es una tecla numérica, el programa modifica el color de CO. Si A$ no es una tecla numérica, el cursor se convierte en A$.

Aparte de esta entrada, el programa sólo responde al botón de disparo.

Si éste ha sido pulsado, se borra la pantalla, y se puede volver a dibujar.



ELEKTROCOMPUTER · PRESENTA SUS NUEVOS PRODUCTOS PARA EL VIC-20 Y EL COMMODORE-64. **DATAMASTER 64** Y **CONTROLADOR · C8**, QUE AMPLIAN LAS POSIBILIDADES DE SU ORDENADOR.

DE VENTA EN DISTRIBUIDORES AUTORIZADOS DE TODA ESPAÑA.

***DATAMASTER 64** _ SOFISTICADA BASE DE DATOS PARA EL C—64, PENSADA PARA TRABAJAR CON LA UNIDAD DE DISCO 1541. SIENDO MUY VERSATIL APROVECHA AL MAXIMO LA CAPACIDAD DE MANIOBRA Y ALMACENAMIENTO. NUMERO DE REGISTROS VARIABLE "EJ. 5000 DE 30 CARACTERES". FORMATEADOS Y COPIAS PROGRAMADAS. SALIDA A IMPRESORA (PARALELO CENTRONICS Y SERIE RS232) CHEQUEO OPERACIONES DISCO. GARANTIA 3 MESES. MANUAL COMPL. EN CASTELLANO — P.V.P. 11.800' PTAS.

***CONTROLADOR · C8** _ CONTROLADOR DE 8 RELES PARA EL VIC-20 Y EL C-64. DE FORMA MUY SENCILLA PODEMOS HACER HASTA 255 COMBINACIONES ENTRE LOS 8 RELES, CON UN CONSUMO DE 1000 W. A 220 VOLT. CADA UNO. CON LO CUAL PODEMOS ACCIONAR TODO TIPO DE LUCES O MECANISMOS. INSTRUC. INCLUIDAS. 3 MESES GARANTIA — P.V.P. 9.800' PTAS.



VIA AUGUSTA - 120 - TEF. (93· 2180699 - BARCELONA - 6

Romance con el VIC

Por Diego Romero.

Hemos recibido numerosas cartas pidiéndonos que dedicásemos algún artículo al modo de utilizar las capacidades musicales del VIC-20 y otras solicitando información sobre los comandos de sonido del Super-Expander. Para intentar satisfacer a todos los lectores que lo han solicitado y como pensamos que el mejor modo de comprender algo es usándolo, aquí presentamos dos programas para demostrar el modo de utilizar el sonido en el VIC.

El programa que aparece en el primer listado se puede ejecutar en cualquier VIC sin ampliación de memoria ni cartucho Super-Expander, se trata de un programa que ejecuta la melodía del ROMANCE ANONIMO. El segundo listado utiliza el cartucho super-expander y reproduce la misma melodía. Como podéis ver el cartucho nos evita tener que realizar gran número de operaciones POKE y utilizar gran cantidad de sentencias DATA, simplificando la obtención de melodías con nuestro micro-ordenador.

Los que tengan oído musical podrán comprobar que el mio no lo es, y si quieren corregir alguna nota que esté equivocada pueden hacerlo, para ello daré una serie de explicaciones sobre los dos programas.

En el primer listado los valores de las notas son los que aparecen en las líneas DATA y deben ser todos mayores que 128 para producir sonido (deben oscilar entre 128 y 255), y los valores que les siguen y oscilan entre 6 y 18 son las duraciones de cada nota. Para introducir silencios debemos hacer que la nota sea 0 y el tiempo deseado. El silencio entre notas lo regula el lazo FOR-NEXT de la línea 60, si aumentamos el número de lazos aumentaremos el tiempo. El lazo FOR-NEXT de la línea 40 es el encargado de regular la duración de las notas, podéis cambiar el número de lazos para cambiar la velocidad de ejecución de la melodía.

En el manual de usuario del VIC-20 tenéis una tabla con los valores que tenéis que utilizar para obtener cada nota si deseáis cambiar alguna nota o componer otra melodía. No obstante la reproducimos aquí para si alguien no tuviera a mano el manual.

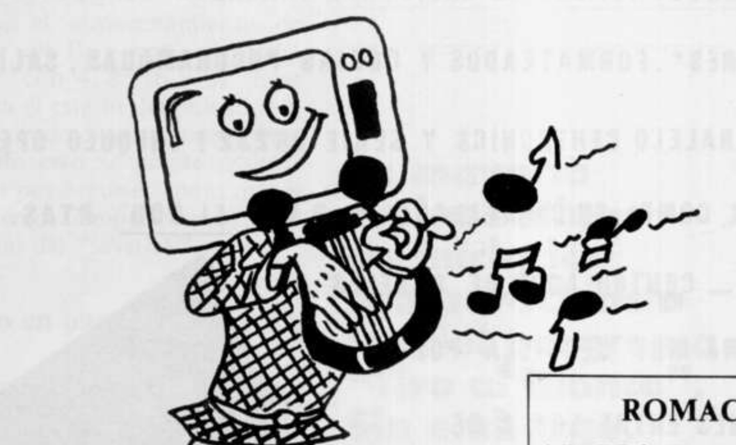
El segundo listado es para el VIC con Super-expander, es mucho más corto, no contiene números misteriosos (salvo algunas excepciones), y las notas se representan directamente. Explicará un poco las cosas raras que aparecen en este listado.

En primer lugar aparece el carácter que activa el sonido con el cartucho, se trata del [CTRL←] en la primera línea PRINT, se obtiene pulsando la tecla de control y la flecha a la izquierda, que es la tecla situada en el extremo superior izquierdo del teclado.

La P indica al ordenador que deseamos ver en la pantalla las órdenes que está ejecutando, esto nos ayudará a ver los efectos de las órdenes que cambiemos.

V9 pone el volumen al máximo, el volumen puede oscilar entre cero y nueve.

S2 indica que queremos utilizar el segundo generador de sonido del VIC-20 (el tenor según el manual). Si cambiamos el número que sigue a la S cambiaremos el generador que utilizamos.



Para VIC-20
con o sin
Super-Expander

ROMACE: CON SUPER-EXPANDER

```
10 PRINT"[CTRL←]PV9S2T603EEEEDCC02BA
A03CEAAAAGFFEDDEFEFE#GFEEDCC";
20 PRINT"02BABBBB03C02BAAAT8AT603#C#
C#C#C02BAA#G#G#GG#G03#F#F#F#F#G#F#FE
EE";
30 PRINT"#F#GAAAA#GG#F#F#F#FED#C#C#C
#CD02BAAAT8AT6"
35 FORN=1TO3000:NEXT
40 RUN
```

ROMANCE: SIN SUPER EXPANDER

```
10 POKE36878,15
20 READN,T:IFN=-1THENRESTORE:GOTO20
30 POKE36876,N
40 FORZ=0TOT*50:NEXT
50 POKE36876,0
60 FORZ=0TO100:NEXT
70 GOTO20
100 DATA231,6,231,6,231,6,231,6,228,
6,225,6,225,6,223,6,219,6,219,6
110 DATA225,6,231,6,237,6,237,6,237,
6,237,6,235,6,232,6,232,6,231,6,228,
6
120 DATA228,6,231,6,232,6,231,6,232,
6,231,6,236,6,232,6,231,6,231,6,228,
6,225,6
130 DATA225,6,223,6,219,6,223,6,223,
6,223,6,223,6,225,6,223,6,219,6,219,
6,219,6
140 DATA219,12,0,2,227,6,227,6,227,6
,227,6,223,6,219,6,219,6,217,6,217,6
,217,6,215,6
150 DATA217,6,234,6,234,6,234,6,234,
6,236,6,234,6,234,6,231,6,231,6,231,
6,234,6
160 DATA236,6,237,6,237,6,237,6,237,
6,236,6,235,6,234,6,234,6,234,6,234,
6,231,6
170 DATA228,6,227,6,227,6,227,6,227,
6,228,6,223,6,219,6,219,6,219,6,219,
18,0,40,-1,-1
```

T6 pone el tiempo de ejecución de cada nota, probar a cambiarlo y ver el efecto sobre las notas que lo siguen.

03 selecciona la tercera octava de las tres de que dispone el VIC, si ponemos 02 usaremos la segunda y con 01 la primera.

Finalmente encontramos las letras C, D, E, F, G, A y B que corresponden a las notas de la escala musical Do, Re, etc.

Para obtener bemoles debemos proceder la letra con el símbolo $, y los sostenidos irán precedidos de#.

Para intercalar silencios entre dos notas pondremos una R entre ellas, esto introducirá un Retardo (silencio o Rest) equivalente a la duración de una nota.

Para que el VIC continúe en el modo musical al pasar al siguiente PRINT es necesario terminar la línea con punto y coma (;), ya que cualquier retorno de carro enviado a la pantalla desconectaría el sonido.

Espero que estos dos programas y las explicaciones sirvan para aclarar un poco el modo de utilizar el sonido en el VIC y cómo aprovechar los comandos del Super-Expander.

Centro COMMODORE (SAKATI, S.A.)

NOTICIAS DE SOFT

LISTA DE PROGRAMAS PARA SU COMMODORE 64
MANUALES EN CASTELLANO

Programas de aplicación		
Superbase 64	disco	22.500 Pts.
Base de datos II	disco	5.000 Pts.
	cinta	4.000 Pts.
Busicalc	disco	8.500 Pts.
	cinta	8.000 Pts.
Control de almacén	disco	35.000 Pts.
Mailing	disco	8.000 Pts.
Lápiz óptico	disco	14.000 Pts.
	cinta	13.500 Pts.
Monitor código máquina	disco	6.500 Pts.
	cinta	6.000 Pts.
Desensambledor	cinta	2.000 Pts.

Programas de utilidades		
Extensión de basic	cartucho	25.000 Pts.
Música 64	disco	5.000 Pts.
	cinta	4.000 Pts.
Gráficos 64	disco	8.000 Pts.
	cinta	7.500 Pts.
Velosak	cinta	2.000 Pts.
Tagsort	cinta	2.000 Pts.
Arrow	cartucho	12.000 Pts.
Compactor	cinta	3.000 Pts.
Superbasic	disco	7.500 Pts.
	cinta	6.900 Pts.

Programas de juegos

Palillos
Caktus
Aterrizaje
Comecocos
King Kong
Laberinto
Comando espacial

Batalla espacial
Precio por cinta 2.000 Pts.
Precio en disco 2.500 Pts.
Ajedrez en cinta 4.000 Pts.
en disco 4.500 Pts.
Jarama 84
Camellos

Araña lunar
Defensa lunar
El lago de los tiburones
El hortelano
La bruja
Guerra de zilones

Centro COMMODORE

SOFTWARE PROFESIONAL

Ardemans, 24 Teléf.: 256 77 94, Télex 42222 CICI E Madrid-28

BOLETIN DE PEDIDO
A enviar a **SAKATI, S. A.** Ardemans, 24 MADRID-28

Ref.	Cdad.	Precio
	TOTAL	

☐ Talon adjunto ☐ Contra Reembolso
Fecha Firma

D
Nº Calle
Codigo Postal Ciudad
Provincia Telefono

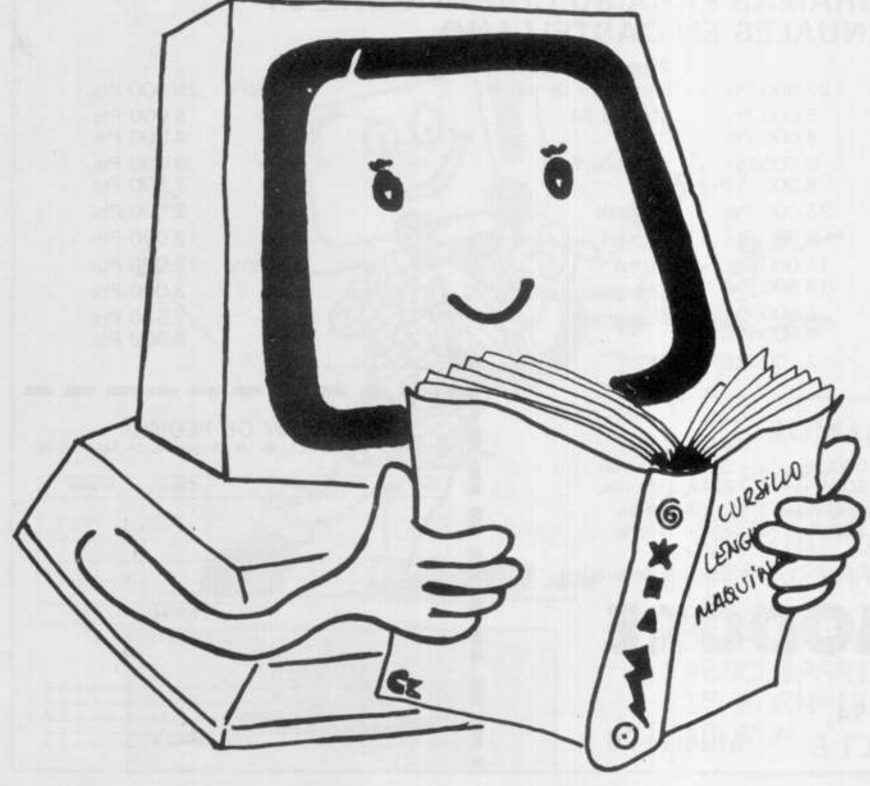
EXCURSION POR LA MICROPROGRAMACION

Cursillo de Lenguaje Máquina —II—

En el número anterior estudiamos los sistemas de numeración que vamos a utilizar más frecuentemente en código máquina, y en este número debemos echar una ojeada al protagonista principal de nuestros desvelos y desdichas en estas lides, como ya habréis imaginado se trata de la CPU.

Por Diego ROMERO

Para comenzar a estudiar las instrucciones de un programa en código-lenguaje máquina, debemos conocer la estructura de los registros sobre los que actuarán, las banderas o flags del registro de estado de la CPU y algo del modo en que opera este chip al ejecutar los pasos de un programa.



Las dos Unidades Centrales de Proceso (6502 y 6510) son muy parecidas, la diferencia reside en que la 6510 tiene un port paralelo de ocho bits incorporado, este port lo estudiamos en "el misterio de los pokes", por tanto nos limitaremos a estudiar la 6502. Esta CPU sirve para un sistema microcomputador de 8 bits de datos, con un bus de direcciones de 16 bits, con él podremos dirigirnos a 2^{16}=65536 direcciones de memoria, esta cantidad se suele llamar 64K.

Estas son las características más importantes de la información del fabricante:

—Alimentación sencilla de + 5V.

—Tecnología de puerta de silicio de canal N.

—Procesamiento paralelo de 8 bits.

—Juego de 56 instrucciones diferentes.

—Aritmética decimal y binaria.

—Trece modos de direccionamiento.

—Capacidad de direccionamiento indexado.

—Stack programable.

—Tres tipos de interrupciones (mascarables, no enmascarables y por programa).

—Rango de memoria direccionable de 64 K (6502) y más de 88 K (6510).

—Puede utilizar cualquier tipo de memoria.

—Capacidad de DMA (acceso directo a memoria).

—Frecuencia de reloj de 1 ó 2 MHz.

—Bus compatible con el sistema 6800 de Motorola.

En la *figura 1* aparece el diagrama de bloques internos de la CPU 6502, es bastante complejo, pero de momento vamos a fijarnos en que está dividido en dos partes: la sección de registros y la sección de control.

La sección de registros se ocupa de todo el procesamiento de datos, mientras que la de control se encarga de descifrar cada una de las 56 instrucciones y generar una secuencia de señales peculiar que obliga a la sección de registros a realizar las operaciones correspondientes a la ejecución de esa instrucción.

Las instrucciones entran en el procesador por el bus de datos, son guardadas temporalmente en el registro de instrucciones y decodificadas por el decodificador de instrucciones.

La unidad de control de tiempos se encarga de asegurar la correcta secuencia de señales para esa instrucción.

REGISTER SECTION | CONTROL SECTION

A0 A1 A2 A3 A4 A5 A6 A7 — ABL — ADDRESS BUS — A8 A9 A10 A11 A12 A13 A14 A15 — ABH

INTERNAL ADL — INTERNAL ADH

INDEX REGISTER Y — INDEX REGISTER X — STACK POINT REGISTER (S) — ALU — ACCUMULATOR A — PCL — PCH — INPUT DATA LATCH (DL) — DATA BUS BUFFER

RES IRQ NMI — INTERRUPT LOGIC — INSTRUCTION DECODE — RDY — TIMING CONTROL — PROCESSOR STATUS REGISTER P — CLOCK GENERATOR — ϕ_0(IN) — ϕ_1 OUT — ϕ_2 OUT — R/W — DBE — INSTRUCTION REGISTER

D0 D1 D2 D3 D4 D5 D6 D7 — DATA BUS

LEGEND: = 8 BIT LINE; = 1 BIT LINE

Al programar en lenguaje máquina es necesario conocer los registros internos de nuestra CPU, son los que aparecen en la *figura 2.* El acumulador A, los registros índices X e Y junto con el puntero de stack S y el de estado P son registros de 8 bits, mientras que el contador de programa es de 16 bits.

Se han respetado los nombres en inglés en los dibujos para acostumbrarnos a conocerlos por las siglas que aparecerán en casi todos los programas, libros y artículos que podamos encontrar sobre código máquina, pero para facilitar las cosas a los que no sepan inglés, incluimos una lista de los equivalentes en español:

ABH y ABL (Address Bus High/Low).—Partes alta y baja del bus de direcciones.

Index Register X/Y.— Registros índices X e Y

Stack Pointer register.—Registro puntero del Stack o pila (S).

ALU.—Unidad Aritmética Lógica, se encarga de todas las operaciones aritméticas y lógicas.

Accumulator.—Es el registro principal de trabajo, el acumulador (A).

PCH y PCL.—Partes alta y baja del contador de programa.

Input Data Latch (DL).—Registro temporal de datos de entrada.

Data bus buffer.—Es un separador que permite a la CPU manejar más circuitos externos sin sobrecargar sus líneas internas.

Interrupt Logic.—Lógica de interrupción. Establece las prioridades entre los distintos tipos de interrupción.

Instruction Decoder.—Decodificador de Instrucciones, descifra lo que significan éstas y ordena la ejecución del procedimiento necesario.

Processor Status Register (P).—Registro interno de estado de la CPU, sirve para saber cuándo un resultado es negativo, demasiado grande, etc.

Instruction register.—Registro de instrucciones, es donde se almacenan mientras se ejecutan.

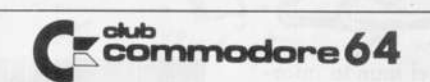
Timing Control.—Control de tiempos.

Clock generator.—Generador de reloj, genera la señal necesaria para controlar todas las transferencias de registros y operaciones de la CPU.

La función del registro de estado de la CPU es la de saber cuándo un resultado fue negativo, cero, demasiado grande, etc., para poder tomar decisiones en función del resultado de la última operación que realizó el procesador. Sus banderas son:

—Carry (C).—bit 0, indica si arrastramos una unidad de la operación anterior (ej. sumando), (1=verdad).

—Zero (Z).—bit 1, indica si el resultado de la última operación fue cero (1=resultado cero).



¿LE PARECE LENTO SU CASSETTE?

Tenemos la solución, necesita nuestro programa FAST-TURBO, multiplica por 10 su velocidad (como una unidad de discos), para Commodore 64.



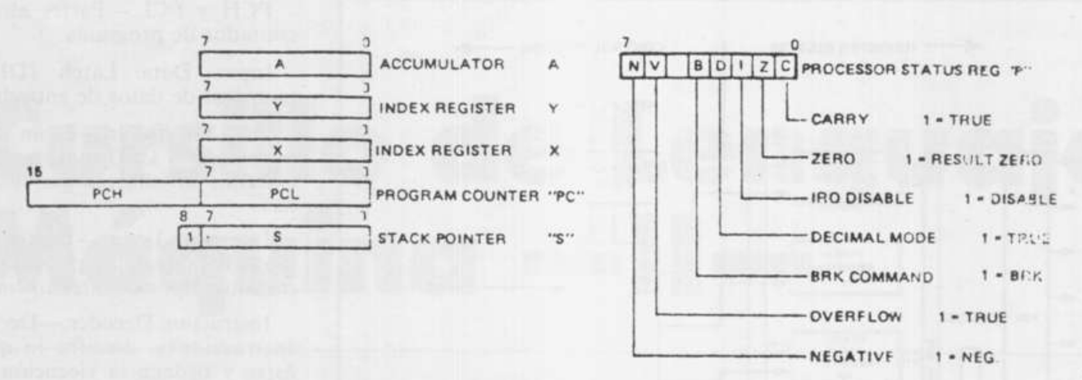


Usted lo lee sólo una vez y leerá todos los programas con gran velocidad

3.500 Pts.

ASTOC-DATA

Hardware y Software-Systems
Sarela de Abajo
Santiago de Compostela
Tel. 599533



REGISTROS INTERNOS DE LA C.P.U. (Fig. 2)

—IRQ disable (I).—bit 2, sirve para deshabilitar las interrupciones mascarables (1=deshabilitadas).

—Decimal mode (D).—bit 3, indica si la CPU trabaja en modo decimal o binario (1= decimal)

—BRK command (B).—bit 4, indica si una interrupción ha sido producida por una instrucción BRK (1=BRK).

—Overflow (V).—bit 6, indica si hubo sobrecarga (1=sobrecarga).

—Negative (N).—bit 7, indica si el resultado de la última operación fue negativo (1=negativo).

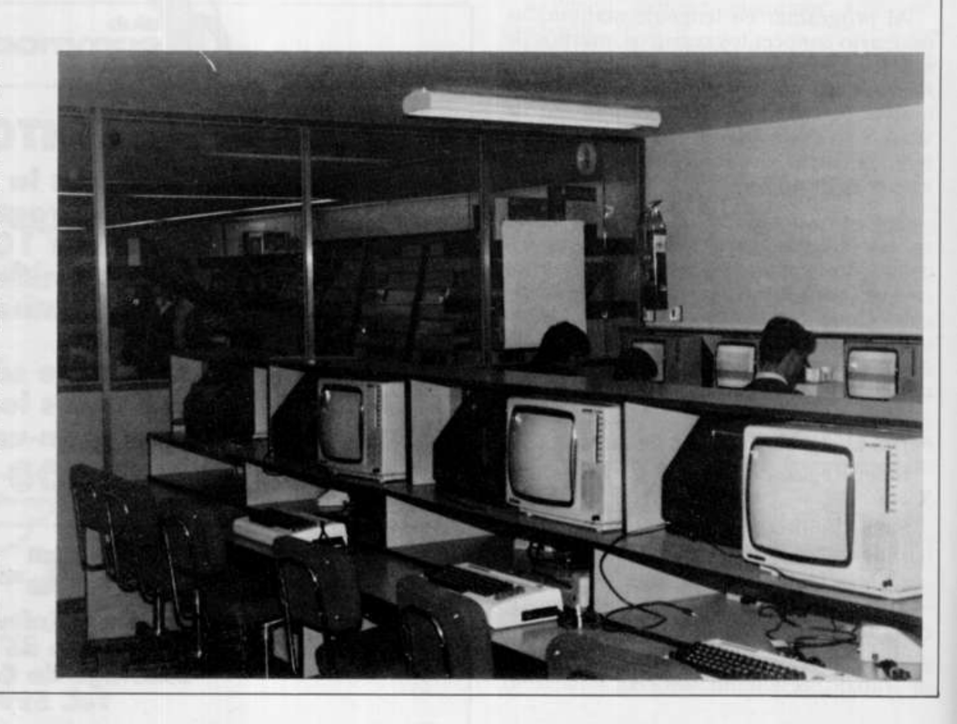
Con este pequeña descripción del interior de nuestro procesador, espero que sea suficiente para poder pasar a la próxima etapa que será... lenguaje máquina!!!

NOTICIAS COMMODORE WORLD

Tronik inaugura en Barcelona un centro integrado en la enseñanza de informática y venta de ordenadores personales

TRONIK aparece en el mundo informático español con la puesta en marcha de un centro de enseñanza exclusivamente informático, con aulas especialmente diseñadas, un profesorado altamente especializado y cursos modulares para niños y adultos, con el que espera dar respuesta a las exigencias del mismo en el ámbito personal y familiar.

Cuenta con una amplia gama de micros de las marcas más conocidas incluyendo el COMMODORE 64 y el VIC 20, así como con una completa programoteca con gran cantidad de juegos y programas escogidos entre los mejores del mercado, tanto nacional como internacional, y una extensa biblioteca sobre micro-informática.



CREAR UN VICASSO con Caracteres Personalizados

Quieres inyectar más creatividad en tu programación, pero el juego de caracteres del VIC-20 no satisface tus necesidades. La única solución es generar tus propios caracteres. Aquí presentamos unas indicaciones útiles que te ayudarán a diseñar rápida y fácilmente tus propios caracteres para juegos y gráficos.



VIC-20

La mayoría de los programadores que tengan interés en juegos y gráficos, llega a un punto en que el juego de caracteres estandard del VIC-20 no satisface la necesidad de creatividad. Aunque existe una buena selección de caracteres interesantes, cuando un juego requiere naves espaciales u hombrecitos, la mejor solución es diseñar un juego personalizado de caracteres programables.

Traslado de Memoria

Las técnicas básicas son bastante sencillas, pero sí requieren un poco de información de fondo para que resulten fácilmente comprensibles. Para empezar, la memoria de caracteres del VIC-20 se almacena en la ROM, la cual no se puede cambiar. Los caracteres sólo pueden ser cambiados cuando se almacenan en la RAM del usuario. Por lo tanto, para poder crear cualquier carácter nuevo, la memoria de caracteres del VIC primero tiene que ser trasladada a las posiciones limitadas en la RAM que puede ser accedida por el chip de vídeo del VIC-20.

Las posiciones estandard en el VIC no ampliado o el VIC de 3K con ampliación, se encuentran al principio de la memoria del usuario en la posición 7168, 6144 o 5120. La posición 7168 almacenará 64 caracteres; la posición 5120 almacenará todos los 255 caracteres. Se traslada la posición del carácter mediante la operación Poke en la posición 36869 con el código apropiado. (La Tabla 1).

Al elegir una posición del carácter, es importante acordarse de que hay que restar la memoria utilizada por el juego de caracteres de la RAM disponible para

POKE 36869	POSICION MEMORIA ROM	POKE 52 Y 56
240		
253	5120	20
254	6144	24
255	7168	28

Tabla 1. Códigos para trasladar y proteger un juego de caracteres

programación. Mientras que la posición 7168 utiliza solamente 512 bytes, la posición 5120, que permite 255 caracteres, utiliza 2560 bytes, dejando solamente 1024 bytes para el resto del programa. Por este motivo, es importante utilizar sólo los caracteres absolutamente imprescindibles.

Otra consideración importante es que el VIC almacena algunos tipos de variables al principio de la memoria del usuario. Para poder proteger el juego de caracteres de estas variables, hace falta hacer un "Poke" en las posiciones 52 y 56 con el código apropiado. La Tabla 1 presenta los códigos para trasladar y proteger un juego de caracteres.

Si se introduce POKE56,28:POKE52,28: POKE36869,255, la pantalla se llenará de basura. Esto es así porque aunque la posición de caracteres se haya trasladado y protegido, todavía no contiene caracteres. Para que la pantalla vuelva a su estado normal, se teclea POKE36869,240.

La siguiente rutina hará un "Peek" en la posición estandard de caracteres y trasladará los 64 caracteres a la nueva posición 7168.

```
10 POKE56,28:POKE52,28:CLR
20 FOR T=7168 TO 7679:POKE T,PEEK
   (T+(32768-7168)):NEXT
30 POKE36869,255
```

Cuando esto se introduce, el único cambio evidente es que el cursor desaparece. Esto es así porque el número del "Poke" en pantalla del espacio invertido que utiliza el cursor es de 160, y el nuevo juego de caracteres contiene sólo 64 caracteres.

Esta fórmula se puede modificar para trasladar más caracteres, cambiando los códigos en las líneas 10 y 30 para que sean los códigos apropiados para la nueva posición y sustituir la nueva posición por 7168 en la línea 20.

También es posible trasladar caracteres individuales al nuevo juego de caracteres. Se puede utilizar la siguiente fórmula, cuando X es igual al código Poke de pantalla del carácter de la ROM, y Y es igual al código Poke de pantalla del carácter que será sustituido.

```
FOR T=O TO 7:POKE7168+Yª8+T,PEEK
(32768+Xª8+T):NEXT
```

Por ejemplo, si se teclea lo que se presenta a continuación, y se pulsa la tecla X, se imprimirá un signo de ?

```
FOR T=O TO 7:POKE 7168+24ª8+T,
PEEK(32768+63ª8+T):NEXT
```

Diseños Originales

Ahora podemos empezar a diseñar caracteres personalizados. Cada carácter se compone de 64 pequeños puntos en la pantalla. Se necesitan ocho bytes de memoria para almacenar un carácter, cada byte compuesto de ocho interruptores de encendido-apagado llamados bits. Si el bit está encendido, también lo está el punto en la pantalla.

A los ocho bits que componen el byte, se les asigna los siguientes valores, que constituyen las potencias de 2 hasta la séptima: 128, 64, 32, 16, 8, 4, 2, 1. Si se utilizan (bit encendido) o no se utilizan (bit apagado) estos números en todas las combinaciones posibles se obtienen los valores de los bytes de 0-255. La Figura 1 presenta la estructura según los bits de un carácter que representa un transbordador del espacio.

Los números a la derecha de la Figura 1 representan los valores obtenidos mediante la suma de los valores de los bits individuales de cada byte. Para sustituir al a por

```
10 POKE52,58:POKE56,28:CLR:DIMN(7)
20 FORT=7168TO7679:POKET,PEEK(T+2560
0):NEXT
30 POKE36869,255
40 READA:IFA=-1THEN200
50 FORT=0TO7:READB:POKE7168+A*8+T,B:
NEXT
60 GOTO40
200 DATA-1
7990 STOP
8000 PRINTCHR$(147)CHR$(18)"[2SPC]76
543210[2SPC]"
8010 FORT=0TO7:PRINTCHR$(18)TSPC(7)T
:NEXT:PRINTCHR$(18)"[2SPC]76543210[2SPC]
"
8020 POKE36869,240:FORT=38400TO39632
:POKET,6:NEXT:L=7704:TH=32
8030 CH=160:LL=7704:PRINT"F1=PINTA[2SPC]
F3=BORRA":PRINT"ESPACIO=MARCA[SPC]F5
=ENTRA"
8040 POKELL,160:FORT=1TO10:NEXT:POKE
LL,32:FORT=1TO10:NEXT
8050 IFPEEK(197)=39THENCH=160
8060 IFPEEK(197)=47THENCH=32
8070 IFPEEK(197)=17THENY=Y-1:IFY<0TH
ENY=0
8080 IFPEEK(197)=18THENY=Y+1:IFY>7TH
ENY=7
8090 IFPEEK(197)=9THENX=X-22:IFX<0TH
ENX=0
8100 IFPEEK(197)=26THENX=X+22:IFX>15
4THENX=154
8110 IFPEEK(197)=32THENTH=CH
8120 IFPEEK(197)=55THEN8140
8130 POKELL,TH:LL=L+X+Y:TH=PEEK(LL):
GOTO8040
8140 PRINTCHR$(19)CHR$(17);:FORX=0TO
7:FORY=0TO7:FORT=631TO640:POKET,0:NE
```

este carácter, se hace un "Poke" con los valores mencionados antes en las primeras ocho posiciones de la memoria de caracteres de la RAM.

El método estandard para realizar esto utiliza las sentencias de datos de la siguiente forma:

```
 10 READ A:IF A=-1 THEN 100
 20 FOR T=O TO 7: READ B: POKE
    (AªB)+7168+T,B:NEXT
 30 DATA 0,0,0, 128, 192, 254, 255, 0, 0, -1
100 END
```

El primer número de datos es el código Poke de pantalla del carácter que se sustituye. El número -1 le indica al programa que el último carácter ha sido introducido. Si se utiliza más de un carácter, el -1 se utiliza solamente después de la línea de datos del último carácter.

Un método aún más sencillo de producir caracteres personalizados es mediante el uso del generador de caracteres programables. Cuando se ejecuta el programa, se trasladan 64 caracteres en la RAM del usuario, se realiza un "Poke" con cualquier carácter nuevo que haya sido diseñado y se hace una pausa para que el usuario compruebe los caracteres nuevos. Si se introduce CONT, el mismo generador de caracteres aparece en pantalla.

El VIC dibujará una caja con los números de los bytes en los laterales y los números de los bits tanto en la parte superior como en la inferior. Los bits se enumeran de siete a cero para demostrar la potencia de dos que representa el valor de dicho bit. Por ejemplo, 2^7=128, el valor del bit más alto.

El carácter nuevo se dibuja moviendo el cursor mediante la tecla W=arriba, X=abajo, A=izquierda, D=derecha. Mediante la pulsación de la barra de espaciado, se coloca una caja coloreada debajo de la posición del cursor. Si se comete un error, se pulsa F3. Esto permite que se borren las cajas coloreadas mediante la pulsación de la barra de espaciado. Si se pulsa F1 el programa vuelve a la modalidad de dibujo.

Cuando se termina de dibujar el carácter, hay que colocar el cursor debajo de la caja y pulsar F5. A continuación, el programa imprimirá los valores Poke para el carácter nuevo y solicitará que se sustituya el código Poke de pantalla de dicho carácter. Ver la guía del usuario en la página 268 del manual. Recuerda también que el juego de caracteres sólo utiliza caracteres de 0-64.

Cuando se introduce este número, el programa escribe automáticamente una línea "DATA" para el carácter nuevo y se suma a si misma esta línea. A continuación, vuelve al principio para introducir el carácter en el juego de caracteres y hace una pausa para que el carácter se pueda comprobar mediante la pulsación de la tecla del carácter que fue sustituido.

En este momento, nunca se debe de usar la tecla "Return", a menos que sea para continuar el programa introduciendo CONT o GOTO8000. Si un carácter no te gusta, puedes hacer uno nuevo y volver a introducir el mismo código de pantalla que antes. Dado que tendrá un número de línea más alto, sustituirá el primer carácter.

Cuando quedan terminados todos los caracteres deseados, se introduce GOTO9000. Esto borra automáticamente la parte del generador del programa, dejando solamente un programa para cargar el nuevo juego de caracteres.

Este programa puede ser salvado en cinta y cargado como cualquier otro. Se pueden añadir juegos antes de la línea 200, o se pueden cargar por separado el juego de caracteres y un juego que lo utilice. Lo único que hay que hacer es dibujar los caracteres; el generador realiza el resto del trabajo.

0	0	0	0	0	0	0	0	=	0
0	0	0	0	0	0	0	0	=	0
1	0	0	0	0	0	0	0	=	128
1	1	0	0	0	0	0	0	=	192
1	1	1	1	1	1	1	0	=	254
1	1	1	1	1	1	1	1	=	255
0	0	0	0	0	0	0	0	=	0
0	0	0	0	0	0	0	0	=	0

Fig. 1 La estructura en bits de un carácter parecido a un transbordador espacial.

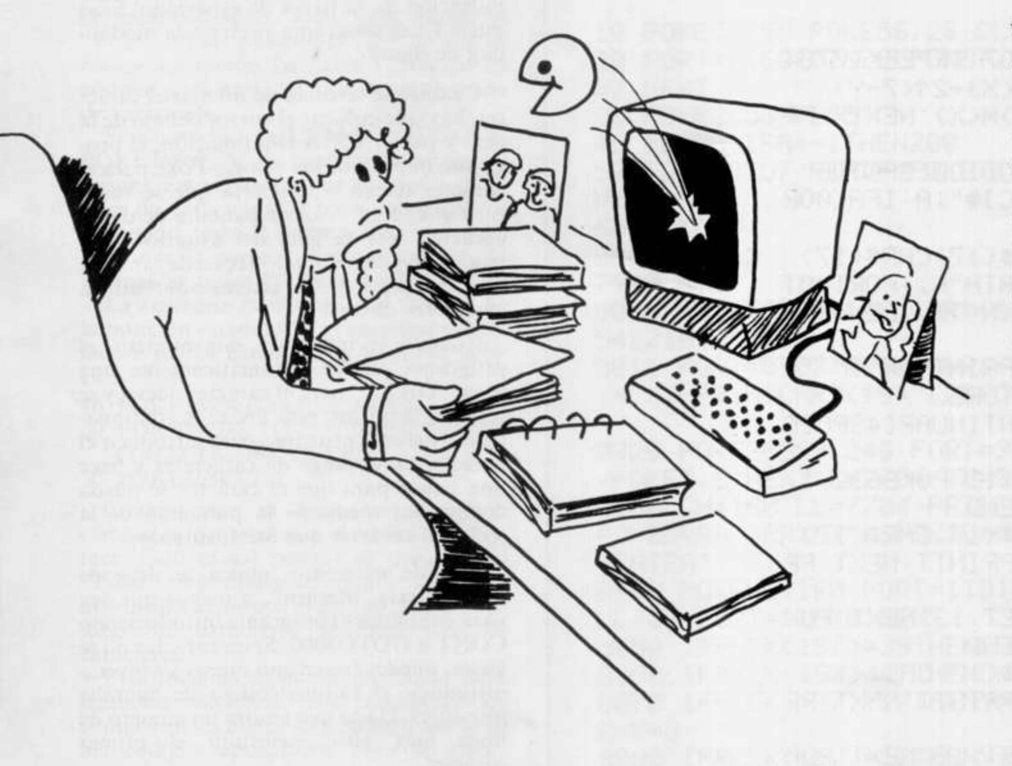
```
XT:NEXT:NEXT
8150 FORX=0TO7:FORY=0TO7:IFPEEK(7704
+X*22+Y)=160THENN(X)=N(X)+2↑(7-Y)
8160 NEXTY:PRINTTAB(14)N(X):NEXTX:PR
INT:PRINT:PRINT
8170 PRINT"ENTRA[SPC]CODIGO[SPC]DE"
8180 INPUT"PANTALLA[SPC]#";A:IFA<0OR
A>64THEN8170
8190 PRINTCHR$(147)CHR$(17)CHR$(17)
8200 Q=Q+1:PRINT60+Q"DATA"A;:FORT=0T
07:PRINT","RIGHT$(STR$(N(T)),LEN(STR
$(N(T)))-1);
8210 NEXT:PRINT:PRINT:PRINT"Q="Q":";
8220 PRINT"GOTO20":RESTORE
8230 PRINT"GOTO8000=CONTINUAR[4SPC]G
OTO9000=END[HOM]"
8240 POKE198,0:POKE631,13:POKE632,17
:POKE633,13:POKE198,3:END
9000 PRINTCHR$(147)CHR$(17)CHR$(17):
FORT=7990TO8070STEP10:PRINTT:NEXT:PR
INT"GOTO9020"
9010 FORT=631TO640:POKET,13:NEXT:POK
E198,10:PRINTCHR$(19):END
9020 PRINTCHR$(147)CHR$(17)CHR$(17):
FORT=8080TO8160STEP10:PRINTT:NEXT:PR
INT"GOTO9040"
9030 FORT=631TO640:POKET,13:NEXT:POK
E198,10:PRINTCHR$(19):END
9040 PRINTCHR$(147)CHR$(17)CHR$(17):
FORT=8170TO8240STEP10:PRINTT:NEXT:PR
INT"GOTO9060"
9050 FORT=631TO640:POKET,13:NEXT:POK
E198,10:PRINTCHR$(19):END
9060 PRINTCHR$(147)CHR$(17)CHR$(17):
FORT=9000TO9070STEP10:PRINTT:NEXT
9070 FORT=631TO640:POKET,13:NEXT:POK
E198,10:PRINTCHR$(19):END
```

Anímate con los "Sprites"

La programación de los gráficos "sprites" no tiene que ser tan laboriosa. Este programa para el C-64 simplifica el proceso.

La capacidad para crear y manipular los gráficos "sprites" es una característica poderosa del Commodore 64. Sin embargo, es un proceso laborioso. Probablemente el aspecto más pesado de los gráficos "sprites" es la traducción de los datos binarios del "sprite" dibujado en números decimales que pueden ser introducidos en la memoria mediante un "Poke".

El programa descrito aquí permite que se dibuje en pantalla una versión ampliada



del "sprite". El ordenador leerá el diagrama, calculará los números que serán introducidos mediante un "Poke" en la memoria y representará el "sprite".

Cómo dibujar un "Sprite"

La guía del usuario del C-64 proporciona una descripción detallada del método para crear un "sprite". En esencia, se rellenan los espacios de un tablero. Un 1 entra en un espacio que será rellenado, y un 0 entra en un espacio que será dejado en blanco. Hay 21 filas y 24 columnas. Las 24 columnas se dividen en tres palabras binarias de 8 bits.

Por lo tanto, las 21 filas, compuestas por tres palabras de 8 bits cada una, componen las 63 palabras que describen el "sprite". Cuando se convierte en valores decimales y se introduce en la memoria mediante un "Poke", el "sprite" puede ser representado en pantalla.

Una vez introducido el programa, se teclea RUN y se dibujan los bordes del "sprite" en pantalla. (No habrá líneas en el tablero). El programa hará una pausa aquí para que el "sprite" pueda ser dibujado dentro de los bordes. Se utilizan las teclas del cursor para que éste se desplace a la posición que el usuario desea rellenar. En esta posición se coloca un 1. No es necesario colocar un 0 en los espacios que quedarán en blanco, ya que el ordenador sólo busca los 1s.

Cuando el dibujo quede terminado, se teclea GOTO200: con el cursor en la mitad del margen izquierdo de la pantalla. (Al teclear estos comandos del programa, es importante incluir los dos puntos. La omisión de uno de ellos resultaría en un error de sintaxis). El dibujo será leído y convertido en decimal, y los valores serán colocados en unas matrices. (Durante unos 20 segundos parece que no está ocurriendo nada). A continuación, se representará el "sprite" tal y como aparecerá en un programa.

El programa vuelve a hacer una pausa, y si el usuario queda satisfecho con el "sprite", puede conseguir una lista de los 63 valores decimales que pueden ser introducidos en la memoria mediante un "Poke" para que el "sprite" pueda ser representado en un programa del mismo usuario. Al teclear GOTO500: la lista aparecerá en pantalla, y GOTO700: enviará la lista a la impresora. Las dos listas se leen de forma horizontal.

Si el usuario no queda satisfecho con el "sprite", y desea modificarlo, también lo puede hacer. Se teclea GOTO200: para que los nuevos valores entren en los "arrays" y se represente la versión nueva.

Al teclear GOTO600: el ordenador volverá a dibujar el "sprite". Para hacer esto, se usarán los datos contenidos en los "arrays". Haga lo que haga el usuario con el dibujo, los datos en el "array" no cambiarán hasta que se teclee GOTO200:. Por supuesto, todo será borrado si se teclea RUN. La Tabla 1 representa un resumen de la función de los comandos GOTO.

C-64

Cómo Funciona el Programa

Este programa funciona dependiendo de la posición del dibujo en la pantalla. Si la pantalla se desplazara incluso una fila hacia arriba, todos los valores para el "sprite" serían incorrectos. Es importante que el cursor no toque la parte inferior de la pantalla; por este motivo los comandos GOTO deben de introducirse a mitad de pantalla. El motivo de que los comandos GOTO van seguidos de dos puntos es para que el ordenador no intente leer la línea entera, la cual incluye parte del dibujo "sprite".

La línea 45 dimensiona los dos "arrays" utilizados y fija la V igual a la dirección del chip controlador de vídeo. Las líneas 100 a 700 llaman las subrutinas que realizan el trabajo del programa.

Las subrutinas 1000 y 1100 hacen los bordes para el dibujo "sprite". La subrutina 1200 lee la zona dentro de los bordes. Si se encuentra un 1, será colocado en el elemento correspondiente del "array" AR. De no ser así, coloca un 0 en la posición del "array". Existen 504 (3×8×21) elementos en el "array" AR.

La subrutina 1300 recoge los datos binarios del "array" AR, ocho elementos por vez, y los trata como si fuese un número binario de 8 bits. Este se convierte en su equivalente decimal y se almacena como una de las 63 palabras en el "array" A1.

La subrutina 1400 representa el "sprite". Los 63 números decimales del "array" A1 se introducen en la memoria mediante un "Poke", comenzando en la posición 832. (Las posiciones 828 a 1019 componen la memoria intermedia de E/S de la cinta.) 832 es de 64 × 13, así que con bloques de 64, estos datos se almacenan en el bloque número 13.

En la línea 1450, la posición 2042 es la que apunta a los datos para el "sprite" 2. El 13 se introduce en esta posición dado que los datos se introdujeron en el bloque 13.

La posición V+21 activa los "sprites". En este caso, activa el "sprite" 2 dado que allí se introdujo un 4 mediante un "Poke". (Ver la Guía de Referencia - página 123). La línea 1440 especifica la posición horizontal y vertical del "sprite".

La subrutina 1500 lista los 63 números decimales que pueden ser utilizados para definir un "sprite" en un programa.

La subrutina 1600 saca los datos binarios del "array" AR y vuelve a dibujar la imagen en pantalla para que pueda ser revisada y modificada si así se desea.

Tabla 1.

Resumen de los comandos GOTO en programa para dibujar un "sprite".

Comando	Acción
GOTO200:	Leer el dibujo "sprite", convertirlo en valores decimales y almacenar éstos en un "array". Representar el "sprite" en pantalla.
GOTO500:	Listar en pantalla los 63 valores decimales. Leer las filas en sentido horizontal.
GOTO600:	Volver a dibujar el "sprite" actual.
GOTO700:	Imprimir lista de los 63 valores decimales. Leer las filas en sentido horizontal.

```
20 REM CON RUN :DIBUJA LOS BORDES OR
IGINALES
25 REM CON GOTO 200:GUARDA DIBUJO Y
SPRITE
30 REM CON GOTO 500:LISTA DATOS EN P
ANTALLA
35 REM CON GOTO 600:VOLVER A DIBUJAR

40 REM CON GOTO 700:LISTA DATOS EN I
MPRESORA
45 DIMAR(504),A1(63):V=53248
100 GOSUB1000
150 END
200 GOSUB1200
300 GOSUB1300
400 GOSUB1400
450 END
500 GOSUB1500
510 IFP1=1THENPRINT#1:P1=0:CLOSE1:RE
M P1=CERO
550 END
600 GOSUB1600
650 END
700 P1=1
710 OPEN1,4:CMD1
720 GOTO500
800 REM
1000 REM PREFIJAR INFORMACION PARA B
ORDES
1010 PRINT"[WHT][CLR]"
1020 S=1034:S1=1057:S2=1:S3=100:GOSU
B1100
```

Continúa en página siguiente



Tele Sant Just
Mayor, 2. Tel. (93) 371 70 43
SAN JUST DESVERN (Barcelona)

INTERFACE para recibir y transmitir CW y RTTY en el VIC-20 y COMMODORE 64

INTERFACE para poder conectar cualquier cassette a los ordenadores COMMODORE 64.

BASE DE DATOS en cassette ARCHIVO DE OSL.

"NECESITAMOS DISTRIBUIDORES"

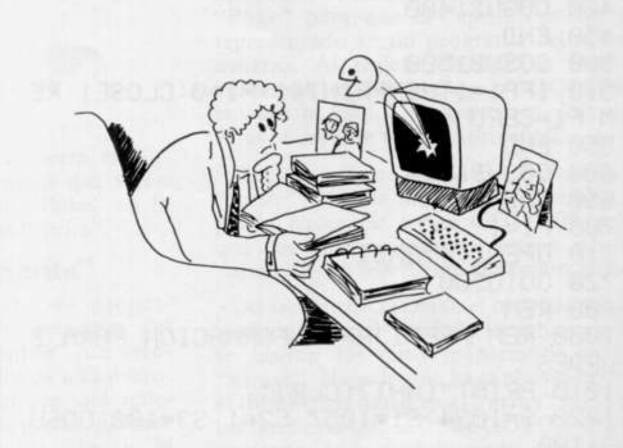
Viene de página anterior

```
1030 S=1073:S1=1873:S2=40:S3=103:GOS
UB1100
1040 S=1914:S1=1937:S2=1:S3=99:GOSUB
1100
1050 S=1098:S1=1937:S2=40:S3=101:GOS
UB1100
1060 S=1034:POKES+7,122:POKES+8,76
1065 POKES+15,122:POKES+16,76
1070 S=1914:POKES+7,80:POKES+8,79
1075 POKES+15,80:POKES+16,79
1080 S=1472:POKES,67:POKES+27,67
1090 RETURN
1100 REM DIBUJA BORDES
1110 FORI=STOS1STEPS2
1120 :   POKEI,S3:POKEI+54272,1
1130 NEXTI
1140 RETURN
1150 REM
1200 REM PONE LOS DATOS DE LOS BIT E
N LA MATRIZ AR()
1210 I1=0:REM I1=CERO
1220 FORS=1074TO1874STEP40
1230   FORI=STOS+23
1240 :     I1=I1+1
1250 :     AR(I1)=0
1260 :     IFPEEK(I)=49THENAR(I1)=1
1270 :   NEXTI
1280 NEXTS
1290 RETURN
1295 REM
1300 REM DECODIFICAR LOS DATOS Y PON
ERLOS EN LA MATRIZ A1()
1310 Y=1
1320 FORI=1TO63
1330 :   DEC=0:BIN=128
1340 :   FORIX=YTOY+7
1350 :     DEC=DEC+BIN*AR(IX)
1360 :     BIN=BIN/2
1370 :   NEXTIX
1380 :   Y=Y+8:A1(I)=DEC
1390 NEXTI
1395 RETURN
1397 REM
1400 REM DIBUJA EL SPRITE
1410 FORI=1TO63
1420 :   POKE831+I,A1(I)
1430 NEXTI
1440 POKEV+4,70:POKEV+5,205
1450 POKE2042,13:POKEV+21,4
1460 RETURN
1470 REM
1500 REM LISTA LOS DATOS
1510 PRINT"[CLR]":POKEV+21,0
1520 FORI=1TO63STEP3
1530 :   PRINTA1(I),A1(I+1),A1(I+2)
1540 NEXTI
1550 RETURN
1560 REM
1600 REM VOLVER A DIBUJAR
1610 GOSUB1000
1620 I1=0
1630 FORS=1074TO1874STEP40
1640 :   FORI=STOS+23
1650 :     I1=I1+1
1660 :     IFAR(I1)=1THENPOKEI,49:PO
KEI+54272,1
1670 :   NEXTI
1680 NEXTS
1690 RETURN
```



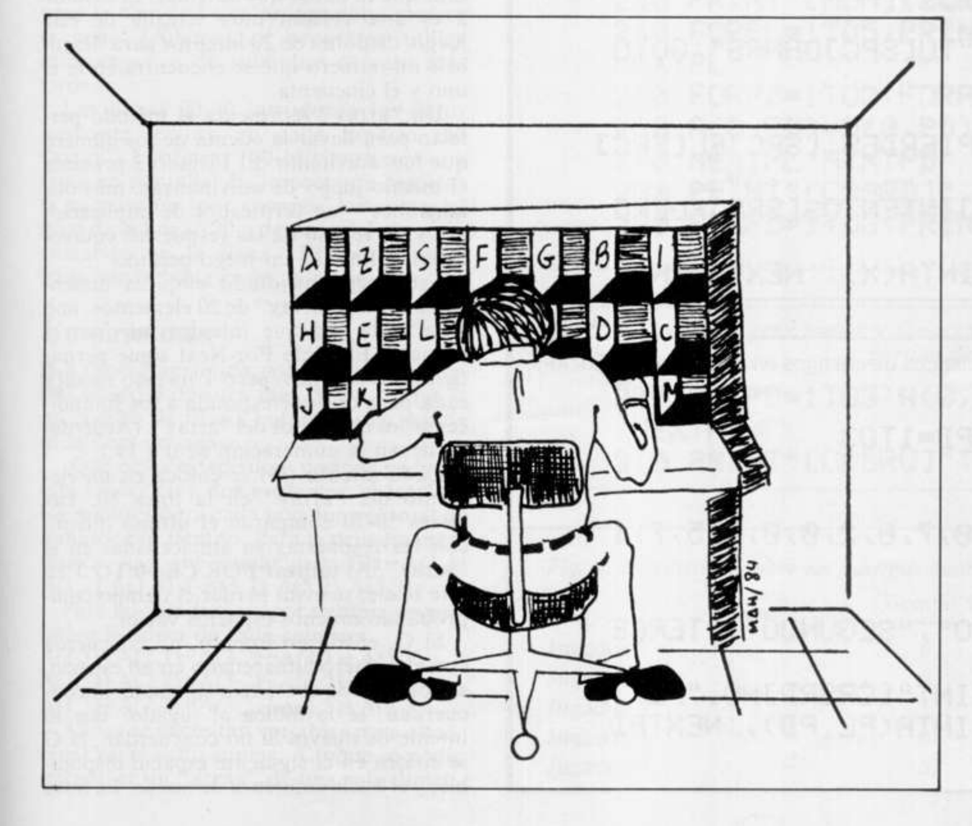

Ejemplar atrasado

250 pesetas



Los "Arrays" a raya

Si tienes ganas de saber lo que son los "arrays" y lo que pueden hacer en tus programas, este artículo intenta aclararte estos puntos oscuros.



La pregunta que más se hace sobre los "arrays", después de "¿Qué son?" es "¿Para qué sirven?" Al principio podrían parecerse a otro tipo de variables; pero sigue leyendo, y descubre algunos de los usos de esta potente técnica de programación.

¿Qué es un "array"?

En los "arrays" se puede almacenar información. Cuando dicha información consiste en números, se utiza un "array" numérico; un "string" de "arrays" se utiliza para almacenar los "strings" de caracteres.

Puedes considerar un "array" como una fila de cajas vacías, cada una capaz de contener cualquier número o "string" que tú le quieres introducir. Cada caja tiene un nombre, compuesto de la variable que da nombre a la fila entera, más un número que indica la posición de la caja en la fila. Si tienes una fila de cinco cajas, podrían llamarse:
A(0), A(1), A(2), A(3), A(4).

El número entre paréntesis se llama un subíndice; A(1) se pronuncia "A sub-uno".

Dimensionar un "array" significa que se le indica al ordenador que guarde un bloque de memoria para almacenar información. Dicha información puede ser proporcionada dentro del mismo programa, o puede ser introducida mientras se está ejecutando el programa.

Para dimensionar un "array", se utiliza la sentencia DIM A (20). Así se le indica al ordenador que guarde 21 espacios de memoria (uno por cada elemento del "array"), el primero siendo A(0), y el último A(20).

El ordenador dimensiona automáticamente un "array" de once elementos, sin emplear la sentencia DIM, si se utiliza una variable con subíndice dentro de un programa. Por ejemplo, si se utiliza 30 A(4)=17, el ordenador dimensionará un "array" llamado A, con los once elementos de A(0) a A(10). Sin embargo, se dimensiona automáticamente sólo si el subíndice es de diez o menos. Para poder utilizar unos subíndices más altos, es imprescindible utilizar primero la sentencia Dim.

Es un buen ejercicio de programación siempre dimensionar un "array", sea cual sea su tamaño; resultará mucho más fácil seguir el listado. También constituye una forma de ahorrar memoria: aunque añade una línea al programa, DIM A(4) dimensiona un "array" de solamente cinco elementos, en vez de los once que se dimensionan automáticamente.

Cualquier variable numérica como A%(4), BP(8), o XL(10) puede ser utilizada como nombre para un "array" numérico.

A veces resulta confuso, y muchas veces poco conveniente, que el Commodore identifique el primer elemento de un "array" como "0". Hay que tener en cuenta que "DIM A (5)" proporciona seis elementos, A(0) a A(5), y no cinco elementos, A(1) a A(5).

Llenando un "Array"

Para llenar una de las "cajas" de un "array" cuando se escribe un programa, se le asigna un valor, igual que se asigna un valor a cualquier otra variable:
A(2)=17:A(4)=8
o
FOR X=1 TO 3: READ A(X): NEXT
DATA 5,17,83

La variable con subíndice que representa un lugar en un "array" puede ser uti-

Listado 1. Programa utilizado para introducir valores en un "array".

```
10 DIMA(5)
15 FORN=0TO5
20 PRINT"QUE[SPC]QUIERES[SPC]EN[SPC]
LA[SPC]CAJA"N
25 INPUTA(N):NEXTN
30 FORN=0TO5
35 PRINT"LA[SPC]CAJA"N"CONTIENE"A(N)

40 NEXTN
```

Listado 2. Una versión sencilla del juego de adivinar números.

```
10 N=INT(RND(0)*50)+1
20 FORT=1TO20
30 INPUT"QUE[SPC]NUMERO";G
40 IFG=NTHENPRINT"TU[SPC]GANAS!":END

50 NEXTT
60 PRINT"TU[SPC]PIERDES.[SPC]EL[SPC]
NUMERO[SPC]ERA"N:END
```

Listado 3. El juego de adivinanzas con unos detalles adicionales.

```
10 DIMA(19)
20 N=INT(RND(0)*50)+1
30 FORT=0TO19
40 INPUT"QUE[SPC]NUMERO";G
50 FORCK=0TOT
60 IFG=A(CK)THENPRINT"YA[SPC]LO[SPC]
HAS[SPC]INTENTADO":GOTO40
70 IFG<>A(CK)THENNEXTCK
80 LETA(T)=G
90 IFG=NTHENPRINT"TU[SPC]GANAS":GOTO
120
100 NEXTT
110 PRINT"TU[SPC]PIERDES.[SPC]EL[SPC]
NUMERO[SPC]ERA"N
120 PRINT"TUS[SPC]INTENTOS[SPC]FUERO
N:":PRINT
130 FORX=0TO19:PRINTA(X),:NEXTX:END
```

Listado 4. Programa para listar las estadísticas de castigos en un partido de hockey.

```
10 DIMA(5,3)
20 FORPL=1TO5:FORPD=1TO3
30 READA(PL,PD)
40 NEXTPD:NEXTPL
50 DATA0,2,2,5,2,0,7,0,2,0,0,2,5,7,1
2
100 PRINT"[CLR]"
110 PRINT,"PRIMERO","SEGUNDO","TERCE
RO"
120 FORPL=1TO5:PRINT"[CRSRD]NO."PL,
130 FORPD=1TO3:PRINTA(PL,PD),:NEXTPD
:NEXTPL
```

lizada como cualquier otra variable. Por ejemplo, A(1)= A(1)+1, PRINT A(3) y PRINT A (4)*16 son todos comandos válidos.

Para llenar una caja durante la ejecución de un programa, lo único que hay que hacer es utilizar el nombre del elemento dentro del comando Input, como se demuestra a continuación: INPUT A(4) o INPUT B(6).

Prueba este programa corto:

```
10 DIM A(2)
20 A(0)=15:A(1)=27:A(2)=32
30 FOR N=0 TO 2: PRINT A(N): NEXT.
```

La línea 10 dimensiona un "array" en tres elementos. La línea 20 llena dichos elementos. La línea 30 inicializa un bucle de modo que la primera vez A(N) es A(0), la segunda vez es A(1), y finalmente es A(2).

Este programa demuestra por qué los "arrays" se utilizan con tanta frecuencia: para la fácil recuperación de la información. Si se hubiera comenzado con A=15: B=27: C=32, no resultaría fácil escribir un bucle que imprimiría los valores de dichas variables.

A propósito, en cuanto se dimensiona un "array" numérico, el ordenador lo llena de ceros, así que un elemento nunca se considera realmente vacío. Los espacios de un "array" de "strings" recien dimensionado contienen "strings" vacíos.

Ahora prueba el programa del Listado 1, que te permitirá introducir los valores para el "array".

Aplicación de los "Arrays"

A lo mejor ya conoces los juegos simples de adivinanza de números, en los cuales el ordenador genera un número al azar que tú tienes que adivinar. El Listado 2 es una versión muy sencilla de este juego: dispones de 20 intentos para descubrir un número que se encuentra entre el uno y el cincuenta.

Un "array" representa el método perfecto para llevar la cuenta de los número que has adivinado. El Listado 3 presenta el mismo juego de adivinanzas, más dos adiciones —un verificador de duplicaciones y un repaso de las respuestas equivocadas al final de un juego perdido.

Este juego mejorado empieza dimensionando un "array" de 20 elementos, uno para cada vez que intentas adivinar el número. El bucle For-Next sigue permitiendo 20 intentos, pero T ha sido modificada para que corresponda a los subíndices de los elementos del "array". (Acuérdate, llevan la numeración de 0 a 19.)

Cada intento (G) se coloca en un elemento del "array" en la línea 80. Las líneas 50-70 comparan el último intento con las respuestas ya almacenadas en el "array". Al utilizar FOR CK=0 TO T en este bucle, se evita perder el tiempo comprobando muchos espacios vacíos.

Si G, el último intento, no concuerda con el número almacenado en un espacio, se comprueba el espacio siguiente. Si concuerdan, se le indica al jugador que lo intente de nuevo. Si no concuerdan, la G se coloca en el siguiente espacio disponible, y el bucle comienza de nuevo. La línea

120 imprime todos los números almacenados en un "array".

Otra Dimensión

La "fila de cajas" es un "array" de una sola dimensión, todas las cajas ordenadas en una sola fila. Si varias cajas se colocaran unas encima de otras lo que tenemos es un "array" de dos dimensiones. (Fig. 1.)

El comando DIM B(2,3) le indica al ordenador que almacene 12 espacios como las cajas de la Figura 1. A continuación, el ordenador almacena tres "filas" de cuatro espacios. Los espacios de la primera fila están asignados B(0,0) a B(0,3). La Figura 2 demuestra los valores de subíndice para los espacios en este "array" bi-dimensional. Evidentemente, las células de memoria del ordenador no están ordenadas de esta forma, pero la imagen sirve para explicarlo.

Los "arrays" de dos dimensiones tienen muchos usos, desde programas de hojas electrónicas hasta el seguimiento de jugadas de un juego. El Listado 4 es un programa que utiliza un "array" de dos dimensiones. La Figura 3 presenta las estadísticas que el programa maneja, es decir, los minutos de castigo para cada uno de los cinco jugadores en los tres tiempos de un partido de hockey.

Podrás ver que la primera línea del programa dimensiona un "array" de un elemento más largo y más profundo que el que parece necesario. Es así para que el usuario pueda ignorar los subíndices de cero y para que los números concuerden de una forma más lógica: A(1,1) es el primer jugador, primer tiempo; A(2,1) es el segundo jugador primer tiempo, etc.

Algunos expertos podrían considerar que este método desaprovecha el espacio, pero la claridad que proporciona merece la pena. (Además, el programa utiliza estos espacios de subíndice de cero más tarde.)

Las líneas 20-40 introducen los datos mediante lectura en cada espacio del "array". Las líneas 100-130 imprimen una tabla en pantalla. Es importante respetar la puntuación (por ejemplo, las comas al final de la línea 120 y después del segundo comando en la línea 130) para poder imprimir la tabla correctamente.

Un paso más

La tabla imprimida por el programa del Listado 4 da otra idea de un "array": un cuadro de filas y columnas, en vez de montones de cajas.

¿Qué otras estadísticas podrías pedir de esta tabla? Hay dos evidentes: el total de los minutos para cada jugador y el total de minutos por tiempo. Para la primera estadística, hay que sumar cada **fila**: A(1,1) A(1,2) A(1,3) a A(5,1) A(5,2) A(5,3).

Para las estadísticas por tiempo, necesitamos la suma de cada **columna**:
A(1,1)+A(2,1)+A(3,2)+A(4,1)+A(5,1)
A(1,2)+A(2,2)+A(3,2)+A(4,2)+A(5,2)
A(1,3)+A(2,3)+A(3,3)+A(4,3)+A(5,3)

Ahora se necesitan variables para almacenar estas ocho sumas. Aunque se podía inicializar un "array" de una sola dimen-

Listado 5. Programa para sumar las filas y columnas en el juego de hockey-castigos.

```
200 FORPL=1TO5:FORPD=1TO3
210 A(PL,0)=A(PL,0)+A(PL,PD)
220 NEXTPD:NEXTPL
230 PRINT"[HOM][2CRSRD]"
240 FORPL=1TO5:PRINTTAB(37);A(PL,0):
NEXTPL
250 FORPD=1TO3:FORPL=1TO5
260 A(0,PD)=A(0,PD)+A(PL,PD)
270 NEXTPL:NEXTPD
280 PRINT"[CRSRD]",
290 FORPD=1TO3:PRINTA(0,PD),:NEXT
```

Listados 4 y 5 combinados.

```
10 DIMA(5,3)
20 FORPL=1TO5:FORPD=1TO3
30 READA(PL,PD)
40 NEXTPD:NEXTPL
50 DATA0,2,2,5,2,0,7,0,2,0,0,2,5,7,1
2
100 PRINT"[CLR]"
110 PRINT,"PRIMERO","SEGUNDO","TERCE
RO"
120 FORPL=1TO5:PRINT"[CRSRD]NO."PL,
130 FORPD=1TO3:PRINTA(PL,PD),:NEXTPD
:NEXTPL
200 FORPL=1TO5:FORPD=1TO3
210 A(PL,0)=A(PL,0)+A(PL,PD)
220 NEXTPD:NEXTPL
230 PRINT"[HOM][2CRSRD]"
240 FORPL=1TO5:PRINTTAB(37);A(PL,0):
NEXTPL
250 FORPD=1TO3:FORPL=1TO5
260 A(0,PD)=A(0,PD)+A(PL,PD)
270 NEXTPL:NEXTPD
280 PRINT"[CRSRD]",
290 FORPD=1TO3:PRINTA(0,PD),:NEXT
```

Listado 6. Programa para sumar y almacenar el total de los minutos de castigo en el partido de hockey.

```
300 FORPD=1TO3:A(0,0)=A(0,0)+A(0,PD)
:NEXTPD
310 PRINT"[CRSRU]"TAB(37);A(0,0)
```

Fig. 3. Estadísticas sobre los castigos durante un partido de hockey.

	Tiempo 1	Tiempo 2	Tiempo 3
Jugador 1	0	2	2
Jugador 2	5	2	0
Jugador 3	7	0	2
Jugador 4	0	0	2
Jugador 5	5	7	12

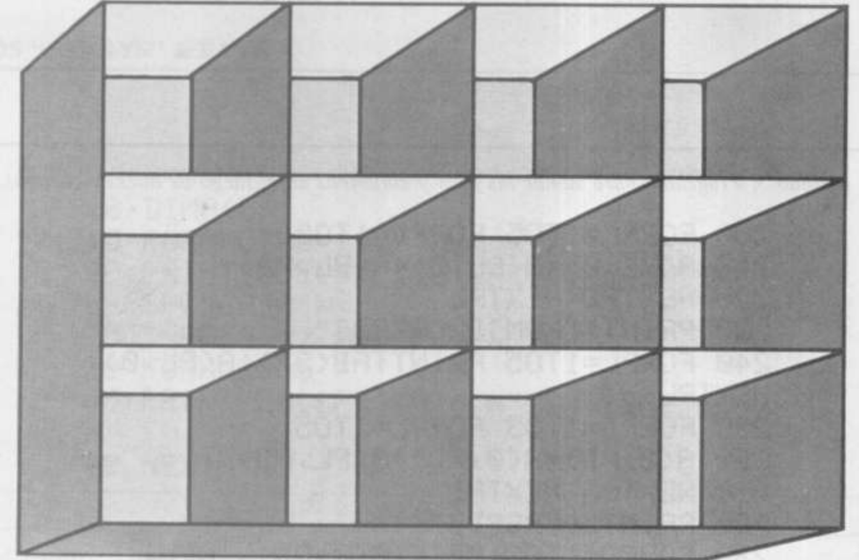


Fig. 1. Una imagen del "montón de cajas" para un "array" de dos dimensiones.

Listados 4, 5 y 6 combinados.

```
10 DIMA(5,3)
20 FORPL=1TO5:FORPD=1TO3
30 READA(PL,PD)
40 NEXTPD:NEXTPL
50 DATA0,2,2,5,2,0,7,0,2,0,0,2,5,7,1
2
100 PRINT"[CLR]"
110 PRINT,"PRIMERO","SEGUNDO","TERCE
RO"
120 FORPL=1TO5:PRINT"[CRSRD]NO."PL,
130 FORPD=1TO3:PRINTA(PL,PD),:NEXTPD
:NEXTPL
200 FORPL=1TO5:FORPD=1TO3
210 A(PL,0)=A(PL,0)+A(PL,PD)
220 NEXTPD:NEXTPL
230 PRINT"[HOM][2CRSRD]"
240 FORPL=1TO5:PRINTTAB(37);A(PL,0):
NEXTPL
250 FORPD=1TO3:FORPL=1TO5
260 A(0,PD)=A(0,PD)+A(PL,PD)
270 NEXTPL:NEXTPD
280 PRINT"[CRSRD]",
290 FORPD=1TO3:PRINTA(0,PD),:NEXT
300 FORPD=1TO3:A(0,0)=A(0,0)+A(0,PD)
:NEXTPD
310 PRINT"[CRSRU]"TAB(37);A(0,0)
```

Listado 7. Programa que demuestra una aplicación de los "arrays" de "strings".

```
10 DIMA$(9)
15 FORN=0TO9:READA$(N):NEXT
20 PRINT"COMO[SPC]TE[SPC]LLAMAS?"
25 INPUTB$
30 FORN=0TO9
35 IFB$=A$(N)THENPRINT"ENTRA":END
40 NEXTN
45 PRINT"VETE":END
50 DATANICOLAS,DANIEL,GLORIA,TOMAS,J
AIME,ANTONIO,MARIA,JOSEFINA,PILI,PED
RO
```

Fig. 2. Valores de subíndice para el "array" de dos dimensiones B(2,3).

0,0	0,1	0,2	0,3
1,0	1,1	1,2	1,3
2,0	2,1	2,2	2,3

sión para almacenarlas, resulta que disponemos de unos espacios vacíos. Se utilizan como se demuestra a continuación:
A(0,1)=Totales del primer tiempo
A(0,2)=Totales del segundo tiempo
A(0,3)=Totales del tercer tiempo
A(1,0)=Totales del primer jugador
A(2,0)=Total del segundo jugador
etcétera.

El Listado 5 se añade al programa. Las líneas 200-220 utilizan los bucles anidados For-Next para sumar las filas de la tabla y almacenar las respuestas de A(1,0) a A(5,0). Las líneas 230 y 240 imprimen los resultados en pantalla. Las líneas 250-290 suman las columnas, almacenan las respuestas de A(0,1) a A(0,3) e imprimen las respuestas en pantalla.

Hay otra estadística que puede ser proporcionada por el programa: el total de los minutos de castigo del juego. A lo mejor te has dado cuenta de que todavía queda un espacio de "array" vacío: A(0,0). Este se puede utilizar para almacenar el total de las columnas (el total de las filas funciona igual de bien). Se agrega el Listado 6 al programa y se vuelve a ejecutar.

Aplicaciones de los "arrays" de "strings"

Los "arrays" de "strings" se utilizan con frecuencia en juegos de pregunta/respuesta. Para simplificar, dos "arrays" de una sola dimensión pueden ser utilizados para que A$ (1) sea la respuesta que acompaña A$ (1). Otra aplicación de los "strings" de "arrays" es para la clasificación en orden alfabético. El tercer uso es para archivar y recuperar registros.

El programa del Listado 7 demuestra este uso: un club privado tiene una puerta de apertura controlada por ordenador que se abre solamente si se proporciona el nombre correcto. El programa funciona casi de la misma forma que la subrutina en el juego de adivinar; compara el nombre dado al nombre almacenado en cada espacio del "array".

Dimensiones múltiples

Los "arrays" de "strings" no están limitados a una dimensión, ni los "arrays" numéricos a dos dimensiones. Es posible utilizar tres, cuatro, cinco o más dimensiones. ¿Qué se puede hacer con ellas?

Por ejemplo, podrías necesitar llevar la cuenta de los minutos de castigo de los jugadores de hockey por tiempo, por juego y por **temporada**. En este caso, necesitas un "array" de tres dimensiones, y puedes visualizar la tercera dimensión (temporadas) como páginas llenas de filas y columnas de números.

Aunque sí existen límites a la operación de dimensionar en el Commodore, aún incluyendo el amplio espacio de memoria, no es probable que rebases la capacidad del equipo.

MANTENIMIENTO DE ORDENADORES COMMODORE

- Desde 1979 damos mantenimiento a los ordenadores Commodore y a sus periféricos, en las series 2000, 3000, 4000, 8000, y 700; con todos sus periféricos además de impresoras Epson, Citoh, Oki, Texas, Seikosha, etc.; discos Commodore, Computhink, Pertec, Tandon, MPI, etc.

- Además, al suscribir un contrato de mantenimiento o contrato de asistencia técnica, le obsequiamos con un seguro a todo riesgo que cubre todas las contingencias, incluso la recuperación de datos.

- Servicios especiales 24 horas/365 días para instalaciones críticas.

Iberdata, S. A. División Mantenimiento de Ordenadores

ALICANTE
(965) 12 09 00
c/ Padre Vendrell, 4
Alicante-3

MADRID
(91) 459 01 50
Pº Castellana, 141
Madrid-16

VALENCIA
(96) 351 19 61
c/ Cervantes, 2 (Guillén de Castro)
Valencia-5

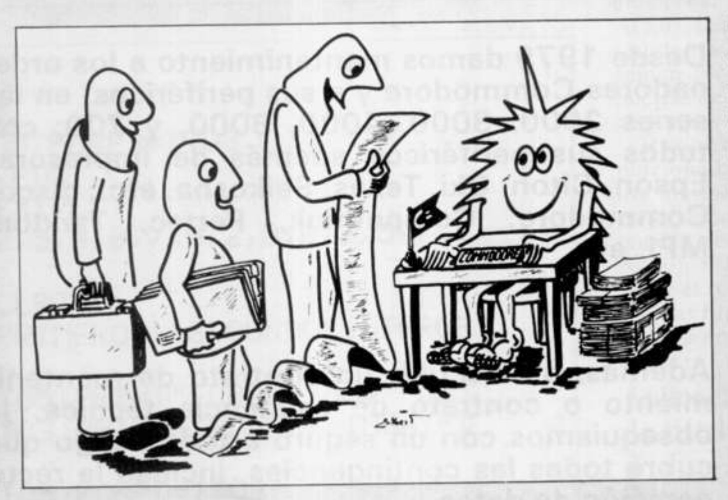
club commodore

Compartiendo Experiencias entre amigos

Esta sección está dedicada a la colaboración de todos nuestros lectores y está dividida en dos partes:
1) Programación: Programas y similares
2) Magia
3) Trucos, sugerencias, etc.

Habrá premios y alicientes "para todos los participantes".

Todas las colaboraciones deben venir escritas a máquina a doble espacio y los programas grabados en cinta (si es posible), o con el listado completo en impresora. Nuestros lectores más jóvenes pueden escribir a mano pero con letra muy clara.



Enviarnos vuestra dirección para que podáis poneros en contacto unos con otros. ¡Animaros, chicos (...y chicas)!

Juego de los Chinos

C-64

Aunque no lo he visto nunca publicado, me cuesta creer que un juego tan popular como éste no tenga ya otras muchas versiones para ordenador más o menos logradas. Antes de comentarlo con detalle, conviene puntualizar algunos aspectos sobre la impresión que adjunto del mismo.

En la línea 20 se dan unas equivalencias indispensables (trébol = corazón; cuadrado = 'Q'). Luego, tras los títulos, se pasa al modo de minúsculas (130), lo que no queda directamente reflejado en la impresión. Para ello, se han subrayado las letras que en el listado original iban (y hay que escribir) en MAYUSCULA. Merece la pena mantener en general el modo de minúsculas, sobre todo para las instrucciones iniciales (130-210), ya que se leen con mayor comodidad, y también para alternarlo con mensajes en mayúscula y dar así más diversidad gráfica a una pantalla relativamente repleta de texto.

El programa ha sido ensayado en un C-64; de ahí que la impresión adjunta ocupe 40 tipos por línea, lo mismo que la pantalla. Eso ayuda muchísimo, en especial para la copia de caracteres iguales seguidos o de expresiones separadas por más de tres espacios. Para el VIC-20, basta redistribuir el espacio de las instrucciones y mensajes subsiguientes.

No hay efectos de color ni de sonido. Se ha procurado, en cambio, que el programa sea robusto y aclaratorio ante cualquier desviación por parte de un usuario ignorante del reglamento o un tanto proclive a las trampas. El juego se desarrolla sólo entre el usuario y el ordenador. Cada jugador sólo conoce la declaración final de su oponente. Ahora bien, el hecho de intervenir una sola pareja hace más vulnerable al jugador que descubre, digamos que involuntariamente, el propio juego al contrincante. Es justo que el ordenador esté apercibido de esta eventualidad (líneas 320-330 y 410-420), en la que, sin embargo, puede incurrir el usuario.

No sería difícil aumentar el programa para hacer que el ordenador se desdoblase en dos o más jugadores, junto o no con el usuario, y estableciera un sistema de eliminación de perdedores.

Comentario detallado del programa

130-210 Entrada al modo de minúsculas, título e instrucciones.
220-240 Acceso al programa.
290... Definición previa de variables de incremento constante: Q1 puntos



```
60 REM
70 REM** R.C. MASSO ** MARZO 1984 **
80 REM
110 REM** INSTRUCCIONES DEL JUEGO **
120 REM
130 PRINTCHR$(14)"[CLR][CRSRD][SPC][SHIFTJ]
[SPC][SHIFTU][SPC][SHIFTE][SPC][SHIFTG]
[SPC][SHIFTO][3SPC][SHIFTD][SPC][SHIFTE]
[3SPC][SHIFTL][SPC][SHIFTO][SPC][SHIFTS]
[3SPC][SHIFTC][SPC][SHIFTH][SPC][SHIFTI]
[SPC][SHIFTN][SPC][SHIFTO][SPC][SHIFTS]"
140 PRINT"[CRSRD][SPC]-------------
--------------------"
150 PRINT"[2CRSRD][SPC][SHIFTI][SHIFTN]
[SHIFTS][SHIFTT][SHIFTR][SHIFTU][2SHIFTC]
[SHIFTI][SHIFTO][SHIFTN][SHIFTE][SHIFTS]
:":PRINT"[SPC][SHIFTI]MAGINATE[SPC]Q
UE[SPC]TU[SPC]Y[SPC]YO[SPC]TENEMOS,[2SPC]
CADA[3SPC]";
160 PRINT"[SPC]UNO,[SPC]TRES[SPC]MON
EDAS[SPC]EN[SPC]EL[SPC]BOLSILLO,[SPC]
Y[SPC]QUE[SPC]PODEMOS[SPC]SACARLAS,";
170 PRINT"[SPC]OCULTAS[SPC]EN[SPC]LA
[SPC]MANO,[3SPC]LAS[SPC]TRES,[SPC]DO
S[SPC]DE[SPC]ELLAS,[SPC]UNA[SPC]O[SPC]
BIEN[SPC]NIN-";
180 PRINT"[SPC]GUNA.[2SPC][SHIFTE]L[SPC]
JUEGO[SPC]CONSISTE[SPC]EN[SPC]ADIVIN
AR[4SPC]CUANTAS[SPC]MONEDAS[SPC]GUAR
DAMOS[SPC]";
190 PRINT"ENTRE[SPC]AMBAS[3SPC]MANOS
,[SPC]TENIENDO[SPC]EN[SPC]CUENTA[SPC]
QUE[SPC]IGNORAMOS[SPC]CUANTAS[SPC]";
200 PRINT"ESCONDE[SPC]EL[SPC]OTRO.":
PRINT"[2CRSRD][SPC][SHIFTH]AREMOS[SPC]
10[SPC]JUGADAS[SPC]EN[SPC]TURNOS[SPC]
ALTERNOS."
210 PRINT"[3CRSRD][SPC][SHIFTC]UANDO
[SPC]LO[SPC]HAYAS[SPC]ENTENDIDO,[SPC]
PULSA[SPC]LA[5SPC]BARRA[SPC]DE[SPC]ESPACIO."
220 GETA$:IFA$=""THEN220
230 IFA$=CHR$(32)THEN290
240 GOTO220
260 REM
270 REM ** FORMATO GENERAL DE LAS BAZAS **
280 REM
290 Q1=0:Q2=0:NR=1
300 FORJ=1TO5:PRINT"[CLR][CRSRD][2SPC]
[SHIFTN]R.";NR;"[3SPC][SHIFTT][SHIFTU]
[SPC][SHIFTT][SHIFTU][SHIFTR][SHIFTN]
[SHIFTO]"
310 PRINT"[SPC]---------------------
-----------------":NR=NR+1
320 GOSUB600:IFB=0THENB2=1:A2=1:GOTO
360
330 IFB1=6THENB2=5:A2=2:GOTO360
340 GOSUB720
350 IFB2=B1ORB2>B1+3ORB2<B1-3THENGOS
UB760:GOTO350
360 PRINT"[CRSRD][SPC][SHIFTY]O[SPC]
DOY[SPC]UN[SPC]TOTAL[SPC]DE";B2
370 GOSUB900
380 PRINT"[CLR][CRSRD][2SPC][SHIFTN]
R.";NR;"[2SPC][SHIFTM][SHIFTI][SPC][SHIFTT]
[SHIFTU][SHIFTR][SHIFTN][SHIFTO]"
390 PRINT"[SPC]---------------------
-----------------":NR=NR+1
400 GOSUB600:GOSUB800
410 IFB2=0THENA2=1:B2=1
420 IFB2=6THENA2=2:B2=5
430 PRINT"[CRSRD][SPC][SHIFTY]O[SPC]
DOY[SPC]UN[SPC]TOTAL[SPC]DE";B2
440 GOSUB700
450 IFB1=B2THENPRINT"[CRSRD][SPC][SHIFTN]
O.[2SPC][SHIFTE]SO[SPC]YA[SPC]LO[SPC]
HE[SPC]DICHO[SPC]YO.[2SPC][SHIFTR]EP
ITE.":GOSUB740:GOTO450
460 GOSUB900:NEXTJ
490 REM
500 REM** RESULTADO GLOBAL **
510 REM
520 PRINT"[CLR][4CRSRD][SPC][SHIFTR]
[SHIFTE][SHIFTS][SHIFTU][SHIFTL][SHIFTT]
[SHIFTA][SHIFTD][SHIFTO][SPC][SHIFTG]
[SHIFTL][SHIFTO][SHIFTB][SHIFTA][SHIFTL]
:":PRINT"[2CRSRD][3SPC][SHIFTT][SHIFTU]
:";Q1:PRINT"[2CRSRD][3SPC][SHIFTY][SHIFTO]
:";Q2
530 PRINT"[7CRSRD][SPC][SHIFTJ]UGAMO
S[SPC]DE[SPC]NUEVO[SPC]([SHIFTS]/[SHIFTN])?"
540 GETE$:IFE$=""THEN540
550 IFE$="S"THEN290
560 IFE$="N"THENPRINTCHR$(142)"[CLR]":END
570 GOTO540
600 REM
610 REM** EXTRACCION DEL USUARIO **
620 REM
630 PRINT"[CRSRD][SPC][SHIFTT]U,[SPC]
CUANTAS[SPC]SACAS[SPC]DEL[SPC]BOLSILLO?"
640 INPUT"[SPC]";A1
650 IFA1<0ORA1>3THENPRINT"[CRSRD][SPC]
[SHIFTN]O.[2SPC][SHIFTH]A[SPC]DE[SPC]
SER[SPC]ENTRE[SPC]'0'[SPC]Y[SPC]'3'"
:GOTO640
660 RETURN
700 REM
710 REM** DECLARACION DEL USUARIO **
720 REM
730 PRINT"[CRSRD][SPC][SHIFTC]UANTAS
[SPC]CREES[SPC]QUE[SPC]HABRA,[SPC]EN
[SPC]TOTAL?"
740 INPUT"[SPC]";B1
750 IFB1<A1ORB1>A1+3THENPRINT"[CRSRD]
[SPC][SHIFTN]O[SPC]ES[SPC]POSIBLE.[2SPC]
[SHIFTR]EPITE.":GOTO740
760 RETURN
800 REM
810 REM** APUESTA DEL ORDENADOR **
820 REM
830 A2=INT(RND(1)*4)+0
840 B2=INT(RND(1)*4)+A2
850 RETURN
900 REM
910 REM** REPRESENTACION DEL RESULTA
DO Y ASIGNACION DE PUNTOS **
920 REM
930 PRINT"[CRSRD][SPC][SHIFTC]OMO[SPC]
YO[SPC]HE[SPC]SACADO";A2;"MONEDAS...
":FORT=1TO1000:NEXTT:SU=A1+A2
940 PRINT"[2CRSRD][SPC][SHIFTE][SHIFTL]
[SPC][SHIFTR][SHIFTE][SHIFTS][SHIFTU]
[SHIFTL][SHIFTT][SHIFTA][SHIFTD][SHIFTO]
[SPC][SHIFTE][SHIFTS]:[SPC]";SU
950 IFSU=B1THENPRINT"[2CRSRD][SPC][SHIFTT]
[SHIFTU][SPC][SHIFTG][SHIFTA][SHIFTN]
[SHIFTA][SHIFTS]!":Q1=Q1+1:GOTO980
960 IFSU=B2THENPRINT"[2CRSRD][SPC][SHIFTG]
[SHIFTA][SHIFTN][SHIFTO][SPC][SHIFTY]
[SHIFTO]!":Q2=Q2+1:GOTO980
970 PRINT"[2CRSRD][SPC][SHIFTN][SHIFTA]
[SHIFTD][SHIFTI][SHIFTE][SPC][SHIFTG]
[SHIFTA][SHIFTN][SHIFTA]"
980 PRINT"[3CRSRD][SPC][SHIFTP]ARA[SPC]
CONTINUAR,[SPC]PULSA[SPC]LA[SPC]BARR
A[SPC]DE[7SPC][SHIFTE][SHIFTS][SHIFTP]
[SHIFTA][SHIFTC][SHIFTI][SHIFTO]"
990 GETZ$:IFZ$=""THEN990
1000 IFZ$=CHR$(32)THENRETURN
1010 GOTO990
```



del usuario; Q2 puntos del ordenador; NR contaje de las bazas.

300-460 Bucle de cinco dobles bazas. El turno del usuario va de 300 a 370. El del ordenador, de 380 a 460.

320_1... Envío a la subrutina 600 para la extracción del usuario.

630-650 Extracción de monedas (sólo entre cero y 3) por parte del usuario (variable A1).

320_2... Envío a la subrutina 700 para la declaración del usuario.

730-740 Apuesta sobre el total de monedas (variable B1).

750... Este total no puede ser inferior a A1 ni superior a A1+3 (esto es, las propias más las tres que puede extraer el adversario.

320_3... Si el usuario da un total de cero y descubre, por tanto, su juego (ya que sólo puede tener cero monedas en la mano), el ordenador hace directamente una apuesta ganadora sacando una moneda (A2) y dando un total de 1 en B2.

330... Si el usuario da un total de 6 descubriendo también su juego (pues sólo puede tener 3 monedas en la mano), el ordenador también hace otra apuesta ganadora sacando 2 monedas (A2) y dando un total de 5 (B2) en 360.

340... Si no se cumple ninguna de las condiciones anteriores, se envía a la subrutina 800 para la apuesta del ordenador.

830... Extracción aleatoria de cero a 3 monedas (A2).

840... Declaración del total entre A2 y 6 (B2).

350... Se evita que el ordenador dé la misma declaración que el usuario o dé una que se diferencie en más de 3 de la de éste. Si así ocurre, se repite sólo la opción de B2 (envío a 840).

360... Respuesta del ordenador.

370... Envío a la subrutina 900 para el resultado y la asignacpuntos.

930... Se muestra la extracción del ordenador (A2). Bucle de retardo para simular el tiempo que tardan los jugadores para abrir la mano y contar el total de monedas.

940... Resultado real (variable SU).

950-970 Asigna los puntos al acertante, si lo hay, o a nadie. Incrementa, en su caso, los puntos del ganador.

980-1010 Vuelta a la siguiente apuesta.

380-390 Cambio de turno. El ordenador declara primero ahora.

400... Envío a la subrutina 600 para que el usuario haga su extracción previa (véase 320_1).
Envío a la subrutina 800 para la apuesta del ordenador (véase 340).

410-420 Evita que el ordenador incurra en el error de apostar a cero o a 6 (compárese con 320-330) cambiando el valor de las variables +1 y a -1, respectivamente.

430... El ordenador hace su apuesta.

440... Envío a la subrutina 700 para la declaración del usuario (véase 320_2).

450... Evita que el usuario apueste lo mismo que el ordenador.

460... Envío a la subrutina 900, que da el resultado y asigna los puntos (véase 370).
Cierra el bucle principal, abierto en 300, para las cinco dobles bazas.

520... Resultado global con los dos últimos valores de Q1 y Q2.

530-570 Vuelta optativa al juego y a los valores iniciales de Q1, Q2 y NR, en caso afirmativo. En caso contrario, vuelta al modo de mayúsculas y final.

R. C. Massó
Barcelona

Dirección omitida por expreso deseo.

Invasión

VIC-20



Bueno pues, ahí va eso por delante, para que lo tratéis como sabéis. Tiempo atrás mandé algunas colaboraciones al Club, varias de ellas no se publicaron (así debían de ser de malas), vamos a ver ahora qué tal en esta nueva etapa.

Para empezar os paso el listado de un programa que he elegido entre los de mi cajón de sastre, aunque como no me va la explicación mejor que lo teclee alguien y mire qué hace.

De paso os diré que por aquí tengo "lanzados" varios programas de los que a su tiempo os hablaré.

Francesc Castanyer P.
C/ Querétaro, 5, E-2.ª, 2º D
Palma de Mallorca, 7
Baleares

```
1 POKE52,26:POKE56,26:CLR:GOTO25:(C)
 SISCOBIT (MALLORCA)
2 POKER,0:POKER-1,0:POKER-2,0:POKER-
3,0:RETURN
3 PRINT"[HOM][RVSON]PUNTOS:"PU:RETUR
N
4 POKEF,32:F=F-22:POKEF+C,3:POKEF,3
5 IFPEEK(F-22)=2THENPOKER-1,240:PU=P
U+1:PC=PC+1:GOSUB3
6 IFPEEK(F-22)=0THENPOKER-1,220:PU=P
U+5:GOSUB3
7 RETURN
8 K=PEEK(203):POKED,32:D=D+(K=31)-(K
=23):IFD<TTHEND=D+22
9 IFD>T+21THEND=D-22
10 POKED+C,0:POKED,1
11 IFK=32THENF=D-22:POKEF+C,2:POKEF,
170:FORE=100TO0STEP-6:POKER-2,E+155:
NEXT
12 GOSUB2:IFF>PTHENGOSUB4
13 IFRND(1)>.9THENGOSUB16:PRINT"[HOM]
[2CRSRD][SPC]@[2SPC]@[SPC]@[3SPC]@[2SPC]
@[SPC]@":POKER-3,140
14 IFRND(1)>.95THENPRINT"[HOM][2CRSRD]
[22SPC]"
15 GOTO8
16 H=INT(RND(1)*PC)+150:IFPC>250THEN
19
17 IFPEEK(P+H)<>32THENPOKER,240:POKE
P+C+H,7:POKEP+H,5:PC=PC+1:RETURN
18 POKER-2,130:POKEP+C+H,4:POKEP+H,2
:RETURN
19 IFPC<PUTHEN23
20 FORE=TTOT+20STEP2:POKEE,2:NEXT
21 POKER-3,0:FORE=2TO6:FORH=0TO21:PO
KET+C+H,E:POKER-1,H*E+129:FORD=1TO40
:NEXTD,H,E:GOSUB3
22 PRINT"[HOM][9CRSRD][RVSON]LA[SPC]
```



```
INVASION[SPC]SE[SPC]HA[SPC]COM-PLETA
DO[SPC]PESE[SPC]A[SPC]TUS[SPC]ES-FUE
RZOS.":G=6:GOTO24
23 GOSUB2:PRINT"[HOM][9CRSRD][RVSON]
LO[SPC]HICISTE[SPC]HUYEN[SPC]TODOS[SPC]
COMO[SPC]CONEJOS":G=12
24 FORE=0TOG:FORH=255TO120STEP-5:POK
ER-1,H:NEXTH,E:GOSUB2:GOSUB35:RUN30
25 PRINT"[CLR][PUR][22COMMP][RVSON][2SPC]
OTROS[SPC]LUNATICOS[2SPC]!!!"
26 PRINT"[CRSRD][CYN]RESISTE[SPC]Y[SPC]
CARGATE[SPC]LOS[SPC]LUNATICOS[SPC]FA
NATICOS[3SPC]QUE[SPC]TE[SPC]ASALTAN"
27 PRINT"[CRSRD][2SPC]CONTROLA[SPC]E
L[SPC]SIEMBRA-MINAS[SPC]CON[SPC]LOS[SPC]
[BLU]CURSORES[CYN]Y[SPC]SUELTA[SPC]L
AS[SPC]CARGAS"
28 PRINT"PULSANDO[SPC][RVSON][BLU]ES
PACIO[RVSOFF][6SPC][CRSRD]ALIENS:[CRSRD]
"
29 PRINT"[PUR][SPC][SHIFTM][SHIFTN][CRSRD]
[2CRSRL][RVSON][COMMV][COMMC][RVSOFF]
[CRSRD][2CRSRL][SHIFTG][SHIFTH][CRSRU]
[CYN]1[SPC]PUNTO[CRSRU][RED][SPC][SHIFTL]
[SHIFT@][CRSRD][2CRSRL][RVSON][COMMV]
[COMMC][RVSOFF][CRSRD][2CRSRL][SHIFTN]
[SHIFTM][CRSRU][CYN]5[SPC]PUNTOS[CRSRD]
":GOSUB35
30 PRINT"[CLR][RED][RVSON]PUNTOS:[SPC]
0":POKE36869,255
31 FORE=7168TO7215:READD:POKEE,D:NEX
T
32 FORE=7424TO7432:POKEE,0:NEXT
33 C=30720:R=36877:POKER+1,8:POKER+2
,25:D=8142:T=D-22:P=7745
34 FORE=DTO8185:POKEE+C,4:POKEE,4:NE
XT:D=T+11:GOTO8
35 PRINT"[CRSRD][BLK][3SPC][RVSON][PUR]
PULSA[SPC]UNA[SPC]TECLA":POKE198,0
36 GETA$:IFA$=""THEN36
37 RETURN
38 DATA66,66,126,24,126,255,60,102,2
4,24,24,60,60,126,126,126
39 DATA129,66,36,24,126,255,165,36,2
4,24,60,60,60,24,24,0
40 DATA255,2,255,8,255,32,255,128,72
,38,148,94,122,41,100,18
```

Basic en español

C-64

Amigos de COMMODORE WORLD: os envío algunas preguntas y sugerencias, junto con un programa para que, si podéis, lo publiquéis.

```
10 PRINT"[CLR][2CRSRD]COPIANDO[SPC]E
L[SPC]BASIC":POKE1,55
20 FORR=40960TO49151:POKER,PEEK(R):N
EXT:R=41117
30 A$="":B$="":N=0:IFR=41372THENR=R+
1
40 N=N+1:IFPEEK(N+R)<128THENB$=B$+CH
R$(PEEK(N+R)):GOTO40
50 B$=B$+CHR$(PEEK(N+R)-128):B=LEN(B
$)
60 PRINT"[2CRSRD]BASIC[SPC]ACTUAL[SPC]
:[4SPC]";B$
70 INPUT"[CRSRD]SUSTITUIR[SPC]POR...
";A$:IFA$="F"THEN130
80 IFA$=""THEN120
90 IFLEN(A$)<>BTHENPRINTB"[SPC]CARAC
TERES,[SPC]POR[SPC]FAVOR":GOTO60
100 IFB>1THENFORN=1TOB-1:POKEN+R,ASC
(MID$(A$,N,1)):NEXT
110 POKEN+R,ASC(RIGHT$(A$,1))+128
120 R=R+N:IFR<41757THEN30
130 POKE1,54
140 READA:FORV=0TO4:READB:POKEA+V,B:
NEXT
150 END:RESTORE:GOTO30
1000 DATA41848,76,73,83,84,79
2000 REM
2010 REM
2020 REM ** GCO 13-3-1984 **
```

HUELGA DE ARTES GRAFICAS Y CORREOS

Debido a la huelga de los trabajadores del sector de Artes Gráficas y Correos, nuestro número anterior (y posiblemente éste número) llegó con retraso a nuestros lectores. Rogamos sepan disculpar este contratiempo ajeno a nuestra voluntad y hacemos votos para que no ocurra lo mismo con próximos números, que también pueden verse involucrados en una nueva amenaza de huelga.



Es un programa sencillo que permite cambiar todo el BASIC y casi todos los mensajes de error; está basado en el artículo "Arquitectura del COMMODORE 64 (II)" que apareció en el nº 13 de CLUB COMMODORE.

Los comandos BASIC están almacenados entre las direcciones 41118 y 41372 y los mensajes de error entre 41374 y 41757. Hay otros mensajes desperdigados por todo el intérprete.

El programa comienza copiando en RAM el intérprete BASIC completo, lo que le lleva **unos 37** segundos; después ofrece uno por uno todos los comandos en el mismo orden en que están almacenados, y a continuación hace lo mismo con los mensajes de error. Unos y otros están almacenados en la memoria en minúsculas con excepción de la última letra, que va en mayúscula (se ha sumado 128 al código).

Si no se quiere cambiar un comando o mensaje, pulsar RETURN. El programa sólo admite palabras de igual longitud que las ya existentes; ignoro si puede alterarse la longitud sin que se produzcan errores de funcionamiento.

Para salir basta con introducir una "F"; después de eso, el ordenador trabajará con la copia modificada del intérprete, lo que se nota inmediatamente porque aparecerá una versión castellanizada del familiar "READY" (esta es la misión de las líneas 140 y 1000). Usando CONT después de haber salido, se puede continuar haciendo cambios. El IF de la línea 30 tiene la misión de saltar el cero que hay en la posición 41373 y que sirve, supongo, como separación entre lo que son comandos y mensajes de error.

Nota: si se crea un vocabulario propio, éste será compatible con todos los programas que ya están grabados en cinta o disco, salvo que contengan alguna variable que está prohibida por el nuevo BASIC. Por ejemplo, al sustituir IF por SI, la variable SI queda prohibida.

Espero que el programa sirva a aquéllos que quieran empezar a hurgar en el interior del ordenador (como me ocurre a mí).

Guillermo Caamaño Olivares
Arabial, 11 - 1º D
GRANADA-3

Y Otra de Marcianitos

VIC-20

Aquest programa és dels de macianitos. La seva extensió és de 1285 bites.

La distribució del programa és la seguenta:

—(0-1) i (1900-1905) inicialitzen les variables i fiquen en pantalla el misatje d'inici del joc.
—(2-10) configura el obni i determina la seva posició arbitrariament.
—(1015-1030) espera les ordres de moviment i tir.
—(1700-1735) crea l'efecta de tir.
—(2000-2025) mou el canó i esborra el rastre.



—(1035) aquesta instrucció fa que cada 15 moviments del canó es produeixi un moviment de l'obni.
—(3000-3020) crea l'efecta d'explosió.
—(3030-3032) resta quiet esperant l'ordre de reinici del joc.

NOTA DE REDACCION: ¡¡Los que deseéis instrucciones en castellano, porque no entendáis catalán, poneros en contacto con Ricard!!

Ricard Sanahuja Moliner
c/ Confederació, 44
Sant Jordi-Cercs (Barcelona)

```
0 M=0:RUN1900
1 B=8174:TI$="000000":POKE36879,8
2 A=INT(RND(1)*242):POKE36878,15:POK
E36876,220
3 Q=0:PRINT"[CLR]"
6 J=7975-A:D=7976-A:C=7997-A:V=7998-
A
7 Z=8018-A:X=8021-A
10 POKEJ,102:POKED,102:POKEC,81:POKE
V,81:POKEZ,78:POKEX,77:POKE36876,0
1015 GETC$
1020 IFC$=","THENM=1
1025 IFC$="."THENM=2
1030 IFC$="↑"THEN1700
1035 G=G+1:IFG=15THEN2
1040 GOTO2000
1700 FORF=1TO22
1705 R=B-(22*F)
1710 POKER,30
1711 POKEZ,99:POKEX,99
1716 IFR=JORR=DTHEN3000
1718 POKE36878,15:POKE36876,250-F:PO
KEZ,78:POKEX,77
1730 NEXTF
1732 POKE36876,0:GOTO2
```

```
1900 PRINT"[CLR][5CRSRD][6CRSRR][SPC]
**OBNI**"
1901 PRINT"[3CRSRD][CRSRR]MOVIMENT[SPC]
DRETA[SPC]'>'"
1902 PRINT"[2CRSRD]MOVIMENT[SPC]ESQU
ERRA[SPC]'<'"
1903 PRINT"[3CRSRD][SPC]PER[SPC]DISP
ARAR[SPC]'↑'"
1904 FORT=1TO6500:NEXT:GOTO1
2000 IFM=2THENB=B+1:POKEB-1,32
2005 IFM=1THENB=B-1:POKEB+1,32
2010 IFB<8165ANDM=1THENB=8164
2011 IFB>8184ANDM=2THENB=8185
2020 POKEB,65
2025 GOTO1015
3000 POKEJ,42:POKEZ,42:POKEV,42:POKE
J+1,42:POKEC,42:POKEX,42
3001 FORT=1TO1800:NEXT
3002 POKEJ,32:POKEC,32:POKED,32:POKE
V,32
3003 FORT=1TO1000:NEXT
3004 POKEJ-24,42:POKEJ-19,42:POKEZ+2
1,42:POKEZ+26,42:POKEJ-22,42
3005 POKE36877,220:FORL=15TO0STEP-1:
POKE36878,L
```



```
3006 FORH=1TO300
3007 NEXTH:NEXTL:POKE36877,0:POKE368
78,0
3008 PRINT"[CLR]":POKEJ-40,42:POKEJ-
47,42:POKEX+39,42:POKEX+46,42
3009 FORT=1TO2500:NEXT
3010 PRINT"[CLR][WHT][3CRSRD][CRSRR]
HAS[SPC]TRIGAT[SPC]"TI$"[SPC]SEGONDS
```

```
"
3020 PRINT"[2CRSRD][SPC]PER[SPC]TORN
AR[SPC]A[SPC]JUGAR[2CRSRR][2CRSRD]PI
TJA[SPC]LA[SPC]TECLA[SPC]'S'":M=0
3030 GETF$
3031 IFF$="S"THEN1
3032 GOTO3030
```

Mastermind

Aquí os mando estas dos colaboraciones.
La primera es el popular juego "MASTERMIND"; consiste en adivinar 4 números elegidos por el ordenador al azar y adivinarlos con los 6 números que te dá. Tendrás que batir el record establecido en 1500. Este record lo tendrás que batir en 3 jugadas.
El segundo es de más habilidad; tendrás que llegar de una base lunar a otra en el menor tiempo posible.
En el transcurso del viaje tendrás que esquivar obstáculos que aparecen aleatoriamente en la pantalla. Cuando hayas conseguido llegar, el ordenador te dirá cuánto combustible has gastado.

VIC-20

LINEAS XXXXXI-48
Se ponen una serie de variables numéricas con valor inicial igual a 0. Estas variables son las que utiliza el ordenador para cargar en ellas los números que él ha elegido al azar. La pantalla se cambia en marco y pantalla a color negro. Se verifica que el ordenador no elija el 0, porque el valor de los números elegidos, se quiere que sea del 1 al 6. Se explica el objetivo del juego en la pantalla.

LINEAS XXXXX48-91
Estas líneas sirven para meter la combinación de números, se comprueba que si el número metido es cero, en tal caso, se produciría un bucle cerrado que no se desbloquearía hasta meter un número distinto de 0. Se controla el número de oportunidades para hallar la combinación correcta. Hay un mecanismo con la print que hace que las sucesivas combinaciones salgan un dibujo de otras ordenadamente.

LINEAS XXXXX100-220
Se comprueba si el número que tú metes es igual al elegido por el ordenador (tanto en valor y en posición). Si esto ocurre se suma 1 a una variable. Si por ejemplo esa variable (g) fuera igual a 3 entonces aparecerían en la pantalla, al lado, de la combinación de números, tres tréboles, que indicarían que hemos acertado tanto en valor como en posición. A continuación se comprueba cuántos números hemos acertado pero sólo en valor no en posición. Si la condición es afirmativa se suma 1 a una variable. Si la variable (h) ese igual a tres significaría que hemos acertado tres números pero sólo su valor y no su posición. Por lo tanto saldría en la pantalla, continuarión de la combinación introducida tres rombos.

LINEAS XXXXX1050-2000.
La línea 1050 controla el número de jugadas, cuando estas llegan a tres la ejecución del programa se pasa a la línea 3040. Se comprueba en qué oportunidad se acierta la combinación a la 5.ª oportunidad, entonces se le suma a la variable X 200 puntos. La variable X acumula los puntos obtenidos durante las tres jugadas, al finalizar estas tres jugadas el ordenador saca por pantalla tu puntuación y el récord, si tu puntuación es mayor que la del récord, entonces el jugador deberá proponerse un récord más alto, así hasta llegar al límite que sería acertar a la 1.ª la combinación en las 3 jugadas. Este récord es prácticamente imposible.

LINEAS XXXXX260-1030
La línea 260 no sirve absolutamente para nada, pero queda bonita ahí. Las líneas 1000-1030 son las encargadas de visualizar los tréboles según los números acertados.

LINEAS XXXXX200-3020
Estas líneas sirven para visualizar el nº de rombos en la pantalla según el número de aciertos de los números en valor solamente. Algunas poseen música. Mensaje de muy mal.

LINEAS XXXXX3400-4030
Comprueba si has superado el récord. Imprime récord y puntuación en la línea 4030 indica el final.

POSDATA
La emoción no está en hallar la combinación sino en superar el récord; esto le obliga forzosamente a no cometer ningún error y hacer siempre la jugada más óptima.

Programador:	**José Vicente López Navarro**
Edad:	**18 tacos.**
Ordenador:	**VIC-20**
Dirección:	**Francisco Pizarro nº 41**
Ciudad:	**Albacete**
Hobby:	**Informática**
Teléfono:	**967-22 69 67**
Color de los ojos:	**Azul**
Lenguaje preferido:	**BASIC**
Comida preferida:	**No lo sé.**

Espero que os guste. ¡Adiós!



"Viaje Lunar"



José Vicente López Navarro

Lo siento pero no me quedan fuerzas para seguir, aquí os doy un breve resumen:

Las primeras líneas te dan las explicaciones de juego, las siguientes dibujan el paisaje lunar, las otras siguientes controlan si hay choque o no de la nave con los obstáculos, al final el ordenador te dice los petrolones que has gastado (este es el combustible de la nave, los petrolones).

```
1 POKE36879,8:Y=0:R=1500:W=0
30 C=0:D=0:N=0:A1=0:B1=0:C1=0:D1=0
40 A=INT(RND(1)*6)+1:B=INT(RND(1)*6)
+1:C=INT(RND(1)*6)+1:D=INT(RND(1)*6)
+1
43 IFA=BORA=CORA=DORB=AORB=CORB=DORC
=DTHEN40
47 PRINT"[CLR]":PRINT"[WHT][4CRSRD]J
UEGO:ACERTAR[SPC]VALOR[SPC]Y[SPC]POS
ICION[SPC]DE[SPC]4[SPC]NUMEROS"
48 PRINT"[3CRSRD]*NUMEROS"1"-"6:PRIN
T"[WHT][2CRSRD][CRSRR]CON[SPC]TRES[SPC]
JUGADAS-SUPERAR-RECORD":FORE=1TO7000
:NEXTE
49 PRINT"[CLR]":PRINT"[RVSON][5CRSRR]
MASTERMIND[RVSOFF]":PRINT"[2CRSRD][CRSRR]
PUEDE[2SPC]EMPEZAR[SPC]:":W=W+1
50 GETA1:IFA1=0THEN50
60 PRINT"[CRSRD]"A1;
70 GETB1:IFB1=0THEN70
71 PRINTB1;
72 GETC1:IFC1=0THEN72
73 PRINTC1;
74 GETD1:IFD1=0THEN74
75 PRINTD1;
90 N=N+1:IFN>8THEN3000
91 PRINT"[5CRSRL]"
95 G=0
100 IFA=A1THENG=G+1
110 IFB=B1THENG=G+1
120 IFC=C1THENG=G+1
130 IFD=D1THENG=G+1
141 IFG=1THEN1000
142 IFG=2THEN1010
143 IFG=3THEN1020
144 IFG=4THEN1030
148 H=0
149 IFA=B1ORA=C1ORA=D1THENH=H+1
150 IFB=A1ORB=C1ORB=D1THENH=H+1
160 IFC=A1ORC=B1ORC=D1THENH=H+1
170 IFD=A1ORD=B1ORD=C1THENH=H+1
200 IFH=0THEN2000
210 IFH=1THEN2010
220 IFH=2THEN2020
230 IFH=3THEN2030
240 IFH=4THEN2040
260 GOTO55
1000 PRINT"[CRSRU][12CRSRR][SHIFTX]"
:POKE36878,10:POKE36875,215:FORT=1TO
100
1001 NEXTT:POKE36875,0:GOTO148
1010 PRINT"[CRSRU][12CRSRR][2SHIFTX]
":POKE36878,10:POKE36875,231
1011 FORT=1TO120:NEXTT:POKE36875,0:G
OTO148
1020 PRINT"[CRSRU][12CRSRR][3SHIFTX]
":POKE36875,183:FORT=1TO120:NEXTT:
1021 POKE36875,0:GOTO148
1030 PRINT"[CRSRU][12CRSRR][4SHIFTX]
":FORT=1TO26:PRINT"[6CRSRD][RVSON]MU
Y[2SPC]BIEN...[SPC]":NEXTT:
1050 REM MUSICA:IFW=3THEN3040
1060 IFN=1THENX=X+1000
1061 IFN=2THENX=X+500
1062 IFN=3THENX=X+400
1063 IFN=4THENY=Y+300
1064 IFN=5THENX=X+200
1065 IFN=6THENX=X+100
1066 IFN=7THENX=X+50
1067 IFN=8THENX=X+25
1068 W=W+1:IFW=3THEN4000
1069 N=0: GOTO49
2000 POKE36878,10:POKE36875,235:FORT
=1TO150:NEXTT:POKE36875,0:GOTO50
2010 PRINT"[16CRSRR][CRSRU][SHIFTZ]"
:GOTO50
2020 PRINT"[16CRSRR][CRSRU][2SHIFTZ]
":GOTO50
2021 POKE36878,10:POKE36875,159:FORT
=1TO150:NEXTT:POKE36875,0:GOTO50
2030 PRINT"[16CRSRR][CRSRU][3SHIFTZ]
":GOTO50
2040 PRINT"[16CRSRR][CRSRU][4SHIFTZ]
":GOTO50
3000 POKE36878,10:POKE36875,128:PRIN
T"[CLR]":FORT=1TO15:F=0
3010 FORV=1TO 25:PRINT"[7CRSRD][5CRSRR]
[RVSON]MUY[SPC]MAL..[RVSOFF]":FORH=1
TO30:NEXTH:NEXTV
3020 POKE36878,0:IFW=3THEN3040
3030 N=0: GOTO49
3040 PRINT"[CLR]":IFX>RTHENR=X:
4000 IFX>RTHENPRINT"NUEVO[SPC]RECORD
:"X"[SPC]"PTOS."
4010[SPC]PRINT"■■■■■PUNTUACION:"X" P
TOS"
4020[SPC]PRINT"■■■■■RECORD      "R" P
TOS"
4030[SPC]FORB=1TO3000:NEXTB:END
```

```
10 PRINT"[CLR]":POKE36879,9
20 PRINT"[2CRSRD][2CRSRR][WHT]*VIAJE
[SPC]LUNAR*"
```

```
30 PRINT"[CRSRD][CRSRR]DIRIGE[SPC]LA
[SPC]NAVE[SPC]CON[SPC]LAS[SPC]TECLAS
,G(←),H(ABAJO),J(DERECHA).";
```



```
35 PRINT"NO[SPC]TE[SPC]ENTRETENGAS[SPC]
Y[SPC]PROCURA[SPC]NO[SPC]CHOCAR[SPC]
CON[SPC]SUS[SPC]MINAS[SPC]Y[SPC]RAYO
S[SPC]LASERS"
40 PRINT"[3CRSRD][CRSRR][SPC]MUCHA[SPC]
SUERTE[SPC]AMIGO"
42 PRINT"[CRSRR][SPC]----------------
---"
50 FORT=1TO6000:NEXTT:PRINT"[CLR]"
60 FORT=1TO130
70 POKE7680+INT(RND(1)*506),46
80 NEXTT
90 FORT=1TO150
100 POKE7686+INT(RND(1)*506),102
110 NEXTT
130 POKE7724,76:POKE7725,100:POKE772
6,100:POKE7727,100:POKE7728,100:POKE
7729,100
131 POKE7702,78:POKE7703,99:POKE7704
,77:POKE7705,100:POKE7706,100:POKE77
29,77
132 POKE8180,78:POKE8159,111:POKE816
0,111:POKE8161,78:POKE8140,111:POKE8
141,78
133 X=3:Y=0
134 POKE(7680+X+22*Y),86
135 POKE36878,15
136 FORL=130TO254
137 POKE36876,L
138 FORM=1TO40
139 NEXTL
141 POKE36878,0
142 POKE36876,0
150 POKE(7680+X+22*Y),32
151 :POKE(7680+X+22*Y),86
180 GETA$
181 W=W+1
190 IFA$="G"THENX=X-1:POKE36876,143:
FORT=1TO50:NEXTT:POKE36876,0
200 IFA$="H"THENY=Y+1:POKE36876,143:
FORT=1TO100:NEXTT:POKE36876,0
210 IFA$="J"THENX=X+1:POKE36876,183:
FORT=1TO100:NEXTT:POKE36876,0
220 IFPEEK(7680+X+22*Y)=102THEN5000
230 IFPEEK(7680+X+22*Y)=42THEN5000
240 IFPEEK(7680+X+22*Y)=111THEN7000
250 IFPEEK(7680+X+22*Y)=76THEN5000
260 IFPEEK(7680+X+22*Y)=78THEN5000
270 IFPEEK(7680+X+22*Y)=100THEN5000
271 IFPEEK(7680+X+22*Y)=77THEN5000
290 POKE(7680+X+22*Y),86
300 IFA$="J"THENPOKE(7680+(X-1)+22*Y
),32
310 IFA$="H"THENPOKE(7680+X+22*(Y-1)
),32
```

```
320 IFA$="G"THENPOKE(7680+(X+1)+22*Y
),32
330 IFPEEK(7680+(X-1)+22*Y)=46THENPO
KE(7680+(X-1)+22*Y),46
340 IFPEEK(7680+X+22*(Y-1))=46THENPO
KE(7680+X+22*(Y-1)),46
350 X1=INT(RND(1)*22+1):Y1=INT(RND(1
)*22+1)
360 FORG=1TO20: POKE(7680+X1+22*Y1),
42:NEXTG
361 POKE(7680+X1+22*Y1),42
362 POKE(7680+X1+22*Y1),42
363 POKE(7680+X1+22*Y1),42
364 POKE(7680+X1+22*Y1),42
380 POKE36878,15:POKE36876,131:FORG=
1TO40:NEXTG
390 POKE36876,0
500 GOTO180
5000 POKE36877,220
5001 FORL=15TO0STEP-1
5003 POKE36878,L
5004 FORM=1TO900
5006 NEXTM
5008 NEXTL
5009 POKE36877,0
6000 POKE36878,0
6010 FORT=1TO60:NEXTT
6011 PRINT"[CLR]":PRINT"[11CRSRD][6CRSRR]
[RVSON]GAME[SPC]OVER[RVSON]":FORT=1T
O2000:NEXTT:RUN
7000 POKE36878,15
7001 FORL=1TO20
7003 FORM=220-LTO160-LSTEP-4
7004 POKE36876,M
7010 NEXTM
7020 FORM=160-LTO220-LSTEP4
7030 POKE36876,M
7040 NEXTM
7050 NEXTL
7060 POKE36878,0
7070 POKE36876,0
7100 PRINT"[CLR]"
7200 PRINT"[3CRSRD][2CRSRR][4SPC][RVSON]
ENHORABUENA[RVSOFF]":FORT=1TO1000:NE
XTT
8000 PRINT"[RVSON]GASTASTE[CRSRD]"W"
PETROLONES[RVSOFF]"
8010 FORT=1TO3000:NEXTT:PRINT"[CLR]"
:PRINT"[RVSON]OTRO[SPC]JUEGO[SPC](S/
N)?[RVSOFF]"
8020 INPUT"",A$
8030 IFA$="S"THENRUN
8040 IFA$="N"THENSTOP
8050 IFA$=""THEN8010
```




Relación de Tiendas en Madrid donde se Vende ya MicroSistemas

COMPU-SHOP .... **Infanta Mercedes, 89**
COMPUSTORE .... **Doce de Octubre, 32**
COMPUSTORE .... **Sor Angela de la Cruz, 28**
COSESA .... **Barquillo, 25**
DIDISA .... **Pintor Rosales, 26**
ELECTRONICA SANDOVAL .... **Sandoval, 3, 4, 6**
MICROTEC .... **Duque de Sesto, 30**
PEÑA, Libería .... **Cavanilles, 52**
SINCLAIR STORE .... **Bravo Murillo, 2**
VIDEO MUSICA .... **Orense, 28**



Cuando se tiene 64 K de memoria, una magnífica resolución, 16 colores, efectos tridimensionales con sprites, un sonido equivalente al de un sintetizador, un teclado profesional con 62 caracteres gráficos, toda una amplia gama de periféricos, la más completa gama de programas educativos, profesionales y de video-juegos...; en resumen, cuando se es un ordenador personal como no existe ningún otro en el mercado y el más vendido mundialmente, es muy difícil decir sin orgullo que eres un Commodore-64.

Claro que más difícil todavía es decir sin orgullo que tienes un Commodore-64. ¿Por qué no lo comprueba?

COMMODORE 64 LE DA ACCESO A MUCHOS ACCESORIOS

Unidad simple de disco (Monofloppy) 170 K.

Cassette.



Plotter e impresora, 4 colores, 14 c.p.s.

Impresora matricial, tractor, 30 c.p.s.

Interface RS232.

Joy Stick.

Paddle.

Cursos de Introducción al BASIC.

COMMODORE 64 LE MUESTRA PARTE DE SUS PROGRAMAS

Utilitarios y lenguajes

MONITOR LENGUAJE MAQUINA.
FORTH.
LOGO.
PILOT.
MACRO ASSEMBLER.
PROGRAMMER'S UTILITIES.
TURTLE GRAPHICS II.
MASTER.

Sistemas operativos

FILE/BOSS.
CP/M.

Programas de aplicaciones

EASY SCRIPT.
Proceso de texto de gran potencia.

CALC RESULT.
Hoja electrónica de cálculo.

EASY CALC RESULT.
Versión simplificada del CALC RESULT.

MAGIC DESK.
Proceso de texto y gestión de ficheros.

AGENDA TELEFONICA.

Programas educativos

MUSIC MACHINE.
MUSIC COMPOSER.
VISIBLE SOLAR SYSTEM.
SPEED/BINGO MATH.
FISICA I.
MATEMATICAS I.
HISTORIA I.
GEOGRAFIA I.
GEOGRAFIA II.
JUEGOS EDUCATIVOS.
TEMAS MONOGRAFICOS.
CONOCIMIENTOS GENERALES.
QUIMICA I.

Juegos

JUPITER LANDER.
KICKMAN.
SEAWOLF.
RADAR RAT RACE.
TOOTH INVADERS.
LAZARIAN.
OMEGA RACE.
LE MANS.
PINBALL SPECTACULAR.
AVENGER.
SUPERMASH.
FROGMASTER.
GRID RUNNER.
ATTACK OF THE MUTANT CAMELS.
THE PIT.
MR. TNT.
6 GAME PROGRAMS.
BINGO.
ROOTING TOOTING.
MINESSOTA FAT'S POOL CHALLENGE.

... y seguimos ampliando la lista

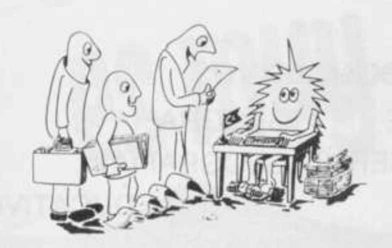
El ordenador personal de la familia más potente

commodore COMPUTER

MICROELECTRONICA Y CONTROL, S.A.
c/. Taquígrafo Serra, 7, 5.°. Barcelona-29
c./ Princesa, 47, 3.° G. Madrid-8

Siglos XX y XXI

PARA VIC-20
SIN AMPLIACION



Ernesto Sánchez, publicó un artículo en el número de febrero de esta revista sobre el uso del Port del Usuario, ahora nos manda un programa para obtener un calendario de cualquier mes de los siglos XX y XXI

Calendario perpetuo de Moret

TABLA I

PARTE DE LA CIFRA DEL AÑO QUE EXPRESA LAS CENTENAS	AÑOS (a)						
	00 (d)	01	02	03		04	05
	06	07		08	09	10	11
		12	13	14	15		16
	17	18	19		20	21	22
	23		24	25	26	27	
	28	29	30	31		32	33
	34	35		36	37	38	39
		40	41	42	43		44
	45	46	47		48	49	50
	51		52	53	54	55	
	56	57	58	59		60	61
	62	63		64	65	66	67
		68	69	70	71		72
	73	74	75		76	77	78
	79		80	81	82	83	
	84	85	86	87		88	89
	90	91		92	93	94	95
		96	97	98	99		
	6	0	1	2	3	4	5
	5	6	0	1	2	3	4
	4	5	6	0	1	2	3
	3	4	5	6	0	1	2
19	2	3	4	5	6	0	1
20	1	2	3	4	5	6	0
	0	1	2	3	4	5	6

TABLA II

II	MESES (d) mayo	ag. febr. B	febr. marzo nov.	junio	set. dic.	abril julio enero B	enero oct.	III	DIAS DEL MES (e) 1 8 15 22 29	2 9 16 23 30	3 10 17 24 31	4 11 18 25	5 12 19 26	6 13 20 27	7 14 21 28
1	2	3	4	5	6	0	1	1	D	L	m	M	J	V	S
2	3	4	5	6	0	1	2	2	L	m	M	J	V	S	D
3	4	5	6	0	1	2	3	3	m	M	J	V	S	D	L
4	5	6	0	1	2	3	4	4	M	J	V	S	D	L	m
5	6	0	1	2	3	4	5	5	J	V	S	D	L	m	M
6	0	1	2	3	4	5	6	6	V	S	D	L	m	M	J
0	1	2	3	4	5	6	0	0	S	D	L	m	M	J	V

1º. Buscar en el cuadro I el número situado en la intersección de la línea que contiene las cifras del siglo y columna que contiene las del año;

2º. Tomar, en la columna exterior del cuadro II, el número así encontrado, y buscar el que se encuentra en la intersección de esta línea y la columna del mes. En los años bisiestos, tomar los meses de enero y febrero marcados con la letra B;

3º. Volver a tomar en la columna exterior del cuadro III, el nuevo número así encontrado. En la intersección de esta línea y la columna de los días del mes se encuentra el día buscado. Ejemplo: ¿Qué día tuvo lugar la batalla de Wagram (6 de julio de 1809)?

Cuadro I: línea 18; columna 09; cifra encontrada 1. Cuadro II: línea 1; columna, julio: cifra encontrada 0. Cuadro III: línea 0; columna 6; día buscado, JUEVES.

Este calendario se presta a búsquedas inversas: buscar los viernes 13, los 1º de enero que caen en un día determinado, etc.

SIGNOS Y ABREVIATURAS: (a) Los siglos gregorianos y los años bisiestos están escritos en cursiva; (b) Hasta el 4 de octubre de 1582 inclusive; (c) Desde el 15 de octubre de 1582 en adelante; las fechas del 5 al 11 de octubre de 1582 no están especificadas en el calendario gregoriano; (d) Los años seculares, siempre bisiestos en el calendario juliano, no lo son en el calendario gregoriano más que cuando sean divisibles por 100; (e) Abreviaturas; m. martes; M. miércoles.



El programa "SIGLOS XX y XXI" presenta por pantalla en formato calendario, un mes cualquiera del año que se desee.

Para ello el programa está basado en las tablas de Moret de cálculo de fechas perpetuas (ver figura 1). Con estas tablas (que están simplificadas para los siglos 20 y 21) se puede calcular qué día de la semana es cualquier fecha que nosotros queramos. Realmente en el programa lo único que se necesita es calcular qué día de la semana es el primer día del mes que escojamos.

De la primera tabla extraemos un dato N1, con el cual en la segunda tabla obtenemos otro número N2 y entonces ya podemos determinar el primer día del mes.

La estructura del programa es la siguiente...

Línea 2-3: subrutina de sonido. Línea 4-9: Datos para calcular el primer número de la Tabla I. Línea 10-40: Presentación. Línea 50-90: Cargar en una matriz de dos dimensiones los datos para el cálculo de N1. Línea 100-310: Solicitud de fecha y tratamiento de la misma. Línea 320-380: Obtención del primer número N1. Línea 400-410: Obtención del segundo número N2. Línea 420-530: Dibujo del calendario. Línea 540-550: Solicitud de nueva fecha.

Variables del programa

X, Y: Variables de bucles FOR, para la matriz de datos de N1
Y%: Año solicitado.
YE%: Variable de año simplificada.
C: Variable dependiente del siglo 20 ó 21.
M1$: Nombre del mes.
M$: Abreviatura de M1$
M: Variable numérica dependiente del mes.
N1: Primer número de la Tabla I
N2: Número de la Tabla II
T: Variable de tabulación para el dibujo del calendario.
D: Variable numérica del día del mes para el dibujo.
D$: Variable de string del día del mes para el dibujo.

Al cuál responderemos con las cuatro cifras, por ejemplo:
AÑO COMPLETO ? 1984 (RETURN)

A continuación se pide el nombre del mes, por ejempleo:
MES (NOMBRE) ? ENERO (RETURN)

En este punto esperamos breves instantes y aparecerá ante nosotros el la pantalla en mes de ENERO del año 1984. Hay que destacar que los domingos aparecen señalados en rojo. Luego el programa solicita una nueva fecha.

Y eso es todo, espero que paséis un buen rato recordando vuestras mejores fechas.

Si me devolvéis la cinta con algo grabado me gustaría que fuera de alguna utilidad. Gracias.

Ernesto Sánchez Gutiérrez'
C./ Mayor, 11
Alcadozo - Albacete


```
PROGRAMA: SIGLOS 20 Y 21

1 GOTO10
2 POKE36875,240:FORK=0TO99:NEXT:
POKE36875,0
3 RETURN
4 DATA04,05,00,01,02,03,100,10,1
1,06,07,100,08,09,100,16,100,12,
13,14,15
5 DATA21,22,17,18,19,100,20,27,1
00,23,100,24,25,26,32,33,28,29,3
0,31,100
6 DATA38,39,34,35,100,36,37,100,
44,100,40,41,42,43,49,50,45,46,4
7,100,48
7 DATA55,100,51,100,52,53,54,60,
61,56,57,58,59,100,66,67,62,63,1
00,64,65
8 DATA100,72,100,68,69,70,71,77,
78,73,74,75,100,76,83,100,79,100
,80,81,82
9 DATA88,89,84,85,86,87,100,94,9
5,90,91,100,92,93,100,100,100,96
,97,98,99
10 POKE36878,15:REM PRESENTACION

20 PRINT"[1CLR][1RVSON][4SPC]SIG
LOS[1SPC]20[1SPC]Y[1SPC]21[4SPC]
[1RVSOF]"
30 PRINT"[4CRSRD]EL[1SPC]PROGRAM
A[1SPC]LE[1SPC]VA[1SPC]A":PRINT"
[1CRSRD]PRESENTAR[1SPC]EL[1SPC]C
ALENDA"
40 PRINT"[1CRSRD]RIO[1SPC]DEL[1S
PC]MES[1SPC]Y[1SPC]A[1RVSON]N[1R
VSOF]O":PRINT"[1CRSRD]DE[1SPC]LO
S[1SPC]SIGLOS[1SPC]20[1SPC]Y[1SP
C]21":PRINT"[1CRSRD]QUE[1SPC]UD.
[1SPC]ELIJA"
50 DIMA%(6,17):REM CARGA DE DATO
S PARA EL 1 NUMERO
60 FORY=0TO17
70 FORX=0TO6
80 READA%(X,Y)
90 NEXTX,Y
100 GOSUB2:PRINT"[1CRSRD]A[1RVSO
N]N[1RVSOF]O[1SPC]COMPLETO":B=0:
REM PETICION DE FECHA
110 POKE198,0:INPUTY%:YE%=Y%
115 IFYE%<1900ORYE%>2099THEN110
120 YE%=YE%-2000:C=2
130 IFSGN(YE%)=-1THENC=1:YE%=100
+YE%
140 IFYE%=100THENYE%=00
150 PRINT"[1CRSRD]MES[1SPC](NOMB
RE)"
160 POKE198,0:INPUTM1$
170 IFYE%/4=INT(YE%/4)THENB=1
180 M$=LEFT$(M1$,3)
190 IFM$="ENE"ORM$="OCT"THENM=0
200 IFM$="FEB"ORM$="MAR"ORM$="NO
V"THENM=3
210 IFM$="ABR"ORM$="JUL"THENM=6
220 IFM$="MAY"THENM=1
230 IFM$="JUN"THENM=4
240 IFM$="AGO"THENM=2
250 IFM$="SEP"ORM$="SET"ORM$="DI
C"THENM=5
260 IFM$="ENE"ANDB=1THENM=6
270 IFM$="FEB"ANDB=1THENM=2
```



Programa siglo XX y XXI (Continuación)

```
280 IFM$="ENE"ORM$="MAR"ORM$="MA
Y"ORM$="JUL"ORM$="AGO"ORM$="OCT"
ORM$="DIC"THENMES=31
290 IFM$="ABR"ORM$="JUN"ORM$="SE
P"ORM$="SET"ORM$="NOV"THENMES=30
300 IFM$="FEB"THENMES=28
310 IFM$="FEB"ANDB=1THENMES=29
315 REM OBTENCION DE N1
320 FORY=0TO17
330 FORX=0TO6
340 IFA%(X,Y)=YE%ANDC=1THENN1=X:
GOTO400
350 IFA%(X,Y)=YE%ANDC=2THENN1=X-
1:GOTO370
360 NEXTX,Y
370 IFX=0THENN1=6:GOTO400
380 IFX<>0THEN400
390 POKE36878,0:STOP
400 N2=N1+M:REM OBTENCION DE N2
410 IFN2>=7THENN2=N2-7
420 REM DIBUJO DEL CALENDARIO
430 PRINT"[1CLR][3CRSRR]A[1RVSON
]N[1RVSOF]O","MES"
440 PRINT"[1CRSRD][1CRSRR]"Y%,M1
$
450 PRINT"[2CRSRD][2CRSRR][1RED]
D[1BLK][2CRSRR]L[2CRSRR]M[2CRSRR
]X[2CRSRR]J[2CRSRR]V[2CRSRR]S[1S
PC]"
460 T=N2*3-2:IFN2=0THENT=19
470 FORD=1TOMES:PRINT"[1BLK][1CR
SRU]"
480 D$=RIGHT$(STR$(D),2)
490 IFT=1THENPRINT"[1RED]"
500 PRINTTAB(T)D$;:GOSUB2
510 T=T+3
520 IFT>21THENT=1:PRINT"[1CRSRD]
";
530 NEXTD
540 PRINT:PRINT"[1BLU]SIGUIENTE[
1SPC]FECHA[1CRSRU]"
550 GOTO100
```

GLOSARIO

Array:
Una colección organizada de datos, los cuales son todos del mismo tipo. Por ejemplo, una lista que consiste en el número de centímetros de lluvia que ha habido en 30 días sucesivos sería un array.
A menudo es más rápido hacer al ordenador procesar datos en forma de arrays en vez de tener al ordenador manteniendo cada dato como elemento separado.

Carácter:
En el proceso de datos, comunicaciones de datos y almacenamiento de información: es cualquier diseño de un código que representa una letra, signo de puntuación, código de control o cualquier otra unidad de información similar. En estas aplicaciones al término "carácter" se utiliza también para valores no imprimibles (tales como los espacios).

Cartucho:
En su aplicación a un cinta magnética es una unidad con varios cientos de pies de cinta (conocido también por el nombre de cartucho de datos) con un tamaño parecido al de un libro de novelas. Usados con la unidad adecuada pueden almacenar varios megabytes.

Chip:
El trozo de material semiconductor en el que se hacen circuitos integrados. Dicho de manera coloquial, un circuito integrado que está empaquetado.

Floppy Disks:
Es un disco magnético en el cual se guardan programas y datos. Los floppy disks se introducen en una unidad de discos para su uso. Su apariencia es la de un disco de 45 rpm. que ha sido dejado encima de un radiador.

Hardware:
Son los elementos visibles. Las partes del ordenador que se pueden tocar, el teclado, la pantalla, los modems. Almacenar información (o retirarla) en el ordenador se hace a través del hardware.

Interface:
Un aparato que permite conectar al ordenador otros aparatos como impresoras u otros ordenadores.

Menú:
Una lista de opciones que aparece en la pantalla del ordenador, describiendo cuáles teclas se han de pulsar para ejecutar ciertas funciones de un programa.

Modem:
Es un periférico. Consiste en una pieza de hardware la cual se conecta a un teléfono convencional logrando que dos ordenadores compatibles puedan intercambiarse información entre sí.

Salida:
Una conexión al mundo exterior de un aparato que puede ser utilizado por los programas o el ordenador.

LE DESCUBRIMOS NUESTRO JUEGO
PARA SU COMMODORE 64

CONTABILIDAD: 300 cuentas, 3.000 apuntes, un programa profesional.
PRACTICALC: hoja de cálculo multifuncional, 2.000 coordenadas.
GESTION STOCK: 1.000 articulos, 1.400 movimientos, completo control de su almacén.
SUPERBASE: potente base de datos, la mejor herramienta.
ULTRABASIC: generador de gráficos y sonidos sin POKES ni PEEKS.
PROCESO DE TEXTOS: calidad en sus escritos.
MAILING: fichero para confección de etiquetas.
VIC-TREE: trabaje con el BASIC 4.0 de COMMODORE.
FILE-BOSS: el mejor auxiliar para trabajar con ficheros relativos.
LAPIZ OPTICO: dibujar directamente sobre su televisor (con software)
JOYSTICK: palanca de juegos para competición, gran sensibilidad.
MONITORES: fosforo verde con y sin sonido.
MATERIAL MAGNETICO: cintas y diskettes, alta calidad.
JUEGOS: en disco, cinta y cartucho, ultimas novedades nacionales y extranjeras.

pidanos información, recibirá un PROGRAMA DE REGALO y nuestro extenso catálogo.

Y AHORA... UN NUEVO SERVICIO!!

sabemos que jugar siempre a lo mismo es muy aburrido...
sabemos que comprar muchos juegos sale caro...
sabemos que... por todo ello, hemos creado...

CLUB TRON DE VIDEOJUEGOS

para COMMODORE - 64

intercambio-alquiler de cintas, discos y cartuchos.
LOS MEJORES PROGRAMAS, LAS ULTIMAS NOVEDADES DEL MERCADO NACIONAL Y EXTRANJERO.

¡PIDA INFORMACION!

EAF microgestion

consejo de ciento, 563-565 barcelona-13
telefono 93-231 95 87 apdo. 24.143

El Misterio de los Pokes

Por Diego Romero

Este mes nos vamos a fijar en las direcciones 0 y 1 del C-64, pero como son bastante complicadas nos limitaremos a lo posible dentro del espacio de que disponemos.

Las funciones de estas direcciones no coinciden con las del VIC-20 como ya hemos visto en el número anterior, y son propias de la CPU 6510 utilizada por el C-64.

Como ya sabréis, el C-64 dispone de 64 K de RAM, 20 K de ROM, y 4 K de direcciones reservadas para entradas/salidas (E/S). La CPU solamente tiene 16 líneas en el bus de direcciones y con ellas sólo puede acceder a 64 K de memoria, para tener acceso a toda su memoria el C-64 utiliza un controlador de memoria ("address manager") que le permite direccionar el total de los 88 K. Podéis programar este controlador de memoria cambiando los contenidos de las direcciones 0 y 1 para obtener las diferentes configuraciones de RAM, ROM y E/S posibles en el C-64.

Estas direcciones se comportan realmente como un port de Entrada/Salida programable, siendo la 1 el registro de E/S de datos (D.R.=Data Register), y la 0 el registro de control de dirección de datos (C.R.=Control Register). Si un bit del registro de control contiene un 1, el bit correspondiente del registro de datos se comportará como una salida, y si el bit del C.R. es 0, el del D.R. será una entrada, por eso es necesario comprobar con PEEK los registros antes de alterar su contenido con POKE para alterar solamente los bit necesarios.

Además del controlador de direcciones, este port controla el DATASSETTE. Las funciones y nombres de cada bit de este port de E/S son los siguientes:

BIT	NOMBRE	DIRECCION	FUNCION
0	LORAM	SALIDA	0=Selecciona RAM en $A000-$BFFF 1=Selecciona ROM en $A000-$BFFF (BASIC)
1	HIRAM	SALIDA	0=Selecciona RAM en $E000-$FFFF 1=Selecciona ROM en $E000-$FFFF (KERNAL)
2	CHAREN	SALIDA	0=Selecciona ROM en $D000-$DFFF 1=Selecciona E/S en $D000-$DFFF
3	CWRITE	SALIDA	Línea de escritura del datassette
4	SWITCH	ENTRADA	Línea del interruptor del datassette
5	CMOTOR	SALIDA	Control del motor del datassette

El valor usual del C.R. será 47 en decimal (%101111), pero cambiará si conectáis algún cartucho.

Como ejemplo del modo de utilizar este port, aquí tenéis el modo de desconectar el intérprete BASIC:

POKE 0, PEEK (0)AND254

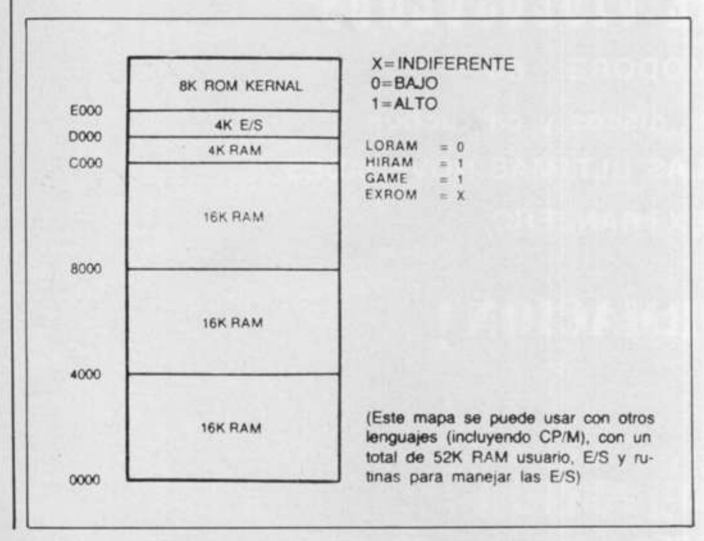
...pero perderéis el control, ya que si desconectáis el basic, no podéis seguir trabajando con él...

Todo esto puede ser muy útil, pero debe utilizarse con mucho cuidado para no perder los programas y casi siempre desde un programa en código máquina.

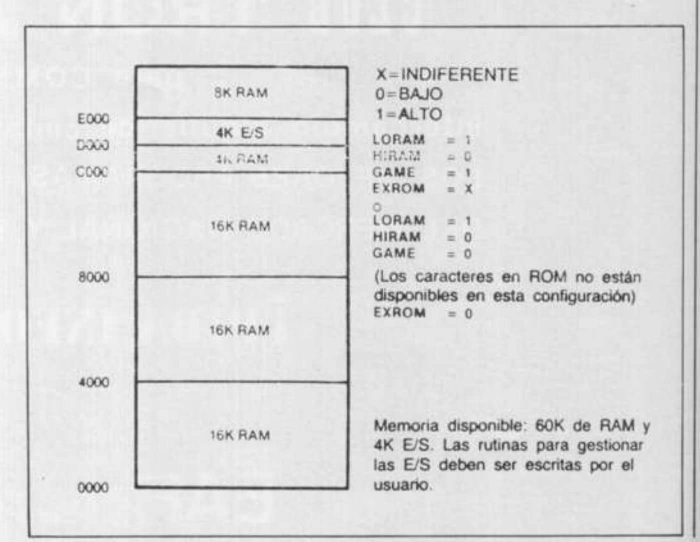
Existen otras dos líneas conectadas al controlador de memoria, se llaman EXROM y GAME, y sirven para que el controlador sepa cuándo tiene conectado un cartucho ROM en el conector de expansión.

Así pues, en función del contenido de la dirección 1 (el D.R.) y de las líneas EXROM y GAME, podemos obtener las configuraciones de memoria en nuestro C-64 que aparecen en las tablas.

La posición de memoria 2 no se utiliza en el C-64. Con esto espero que quede aclarada la función de los POKEs a las tres direcciones del C-64. En el próximo número más "Misteriosos Pokes"...



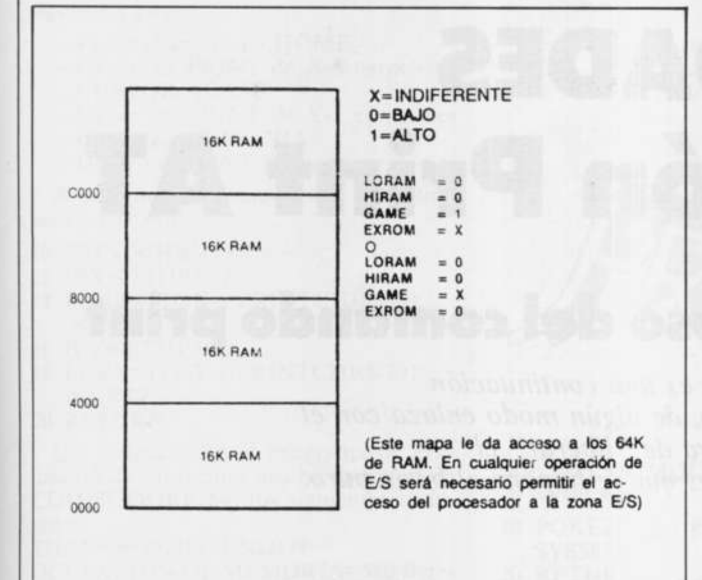
(Este mapa se puede usar con otros lenguajes (incluyendo CP/M), con un total de 52K RAM usuario, E/S y rutinas para manejar las E/S)



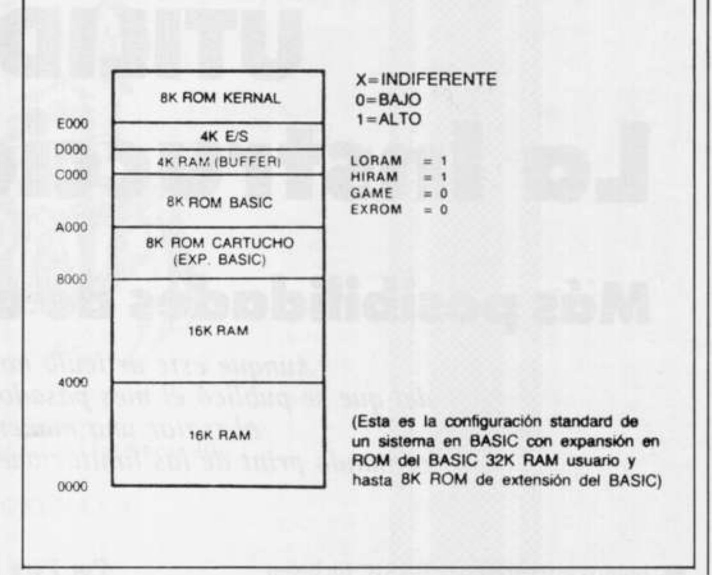
(Los caracteres en ROM no están disponibles en esta configuración)
EXROM = 0

Memoria disponible: 60K de RAM y 4K E/S. Las rutinas para gestionar las E/S deben ser escritas por el usuario.

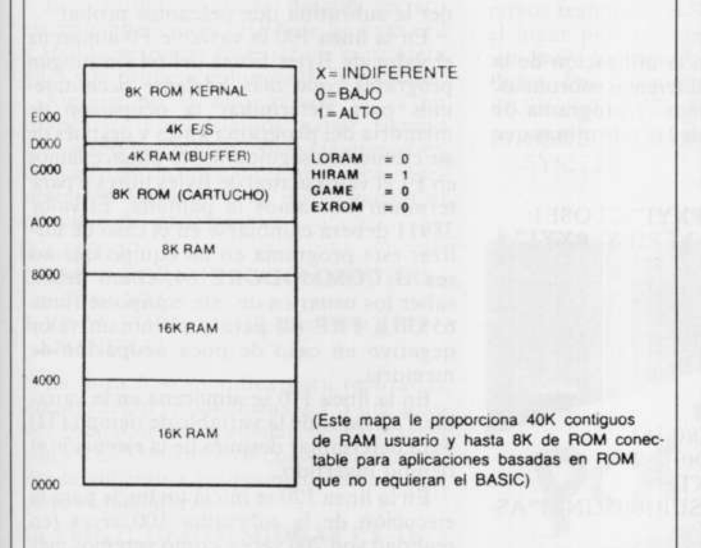




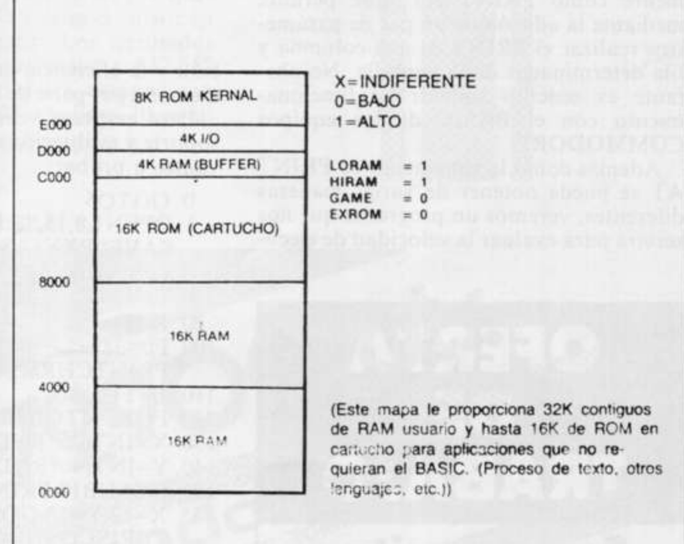
(Este mapa le da acceso a los 64K de RAM. En cualquier operación de E/S será necesario permitir el acceso del procesador a la zona E/S)



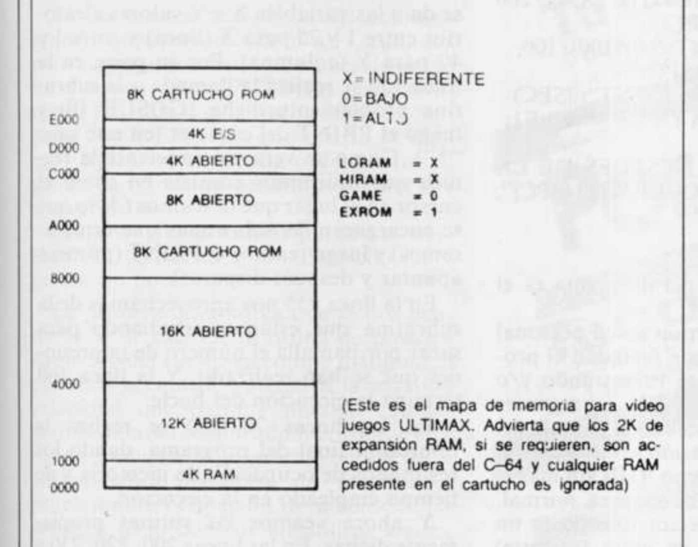
(Esta es la configuración standard de un sistema en BASIC con expansión en ROM del BASIC 32K RAM usuario y hasta 8K ROM de extensión del BASIC)



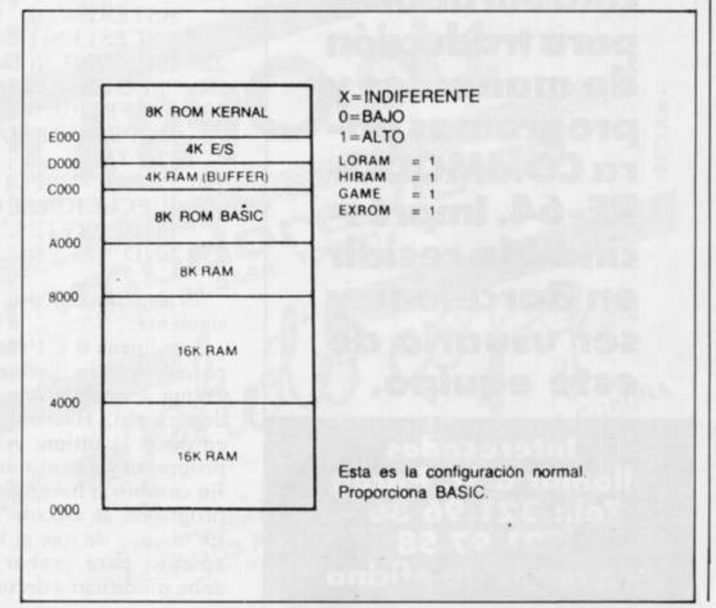
(Este mapa le proporciona 40K contiguos de RAM usuario y hasta 8K de ROM conectable para aplicaciones basadas en ROM que no requieran el BASIC)



(Este mapa le proporciona 32K contiguos de RAM usuario y hasta 16K de ROM en cartucho para aplicaciones que no requieran el BASIC. (Proceso de texto, otros lenguajes, etc.))



(Este es el mapa de memoria para video juegos ULTIMAX. Advierta que los 2K de expansión RAM, si se requieren, son accedidos fuera del C–64 y cualquier RAM presente en el cartucho es ignorada)



Esta es la configuración normal. Proporciona BASIC.



UTILIDADES
La Instrucción Print AT

Más posibilidades de uso del comando print

Aunque este artículo no es una continuación del que se publicó el mes pasado, de algún modo enlaza con él al tratar una manera de "liberar" el comando print de las limitaciones que le impone el basic puro.

Por Pere Masats

Algunos intérpretes de BASIC incluyen un comando que se conoce generalmente como PRINT AT, que permite mediante la adición de un par de parámetros realizar el PRINT en una columna y fila determinadas de la pantalla. No obstante es sencillo simular su funcionamiento con el BASIC de los equipos COMMODORE.

Además como la simulación de PRINT AT se puede obtener de varias maneras diferentes, veremos un programa que nos servirá para evaluar la velocidad de ejecución y la eficiencia en la utilización de la memoria por parte de diferentes subrutinas.

Para empezar veamos el programa de soporte y evaluación de las subrutinas que vamos a probar:

```
  0 GOTO5
  1 OPEN1,8,15,"S:PXY1":CLOSE1:
    SAVE"PXY1",8:VERIFY"PXY1",8
  5 GOTO100
 10 REM
 20 RETURN
100 F0=38911:F1=FRE(0)+65536:
    PRINTCHR$(147);
110 T=TI
120 FORI=1TO100
130 X=INT(25*RND(0))+1
140 Y=INT(40*RND(0))+1
150 GOSUB10:PRINT"*";
155 X=12:Y=13:GOSUB10:PRINTI"AS-
    TERISCOS";
160 NEXTI
200 X=12:Y=1:GOSUB10:PRINT"100
    ASTERISCOS (GOTO X,Y 200
    VECES) EN(13SPC)";
220 PRINTINT (((TI-T)/60)*100)/100;
    "SEGUNDOS(14SPC)"
230 F2=FRE(0)+65536:PRINT"(7SPC)
    EL PROGRAMA OCUPA";F0-F1;
    "BYTES."
240 PRINT "(3SPC) DESPUES DE LA
    EJECUCION SE OCUPAN:(14SPC)";
    F0-F2;"BYTES."
250 END
```

El funcionamiento del programa es el siguiente:

Las líneas 0 y 1 son un truco personal para cargar en disco con facilidad el programa a medida que se va entrando y/o depurando. Haciendo RUN 1 cargamos en disco la última versión (la actual) del programa y a continuación se ejecuta éste. En cambio si hacemos un RUN sin más el programa se ejecuta de manera normal. En el caso de que el lector sólo posea un aparato para grabar en cinta (cassette) debe modificar adecuadamente la línea 1.

La línea 5 nos envía la ejecución a la línea 100 "saltando" por encima de las líneas 10 y 20 que se reservan para contener la subrutina que deseamos probar.

En la línea 100 la variable F0 almacena el valor de Bytes libres del 64 sin ningún programa, que más adelante necesitaremos para determinar la ocupación de memoria del programa antes y después de su ejecución, seguidamente almacenamos en F1 el valor actual de Bytes libres y para terminar borramos la pantalla. El valor 38911 deberá cambiarse en el caso de utilizar este programa en un equipo que no sea el COMMODORE 64, como deben saber los usuarios de este equipo se suma 65536 a FRE (0) para prevenir un valor negativo en caso de poca ocupación de memoria.

En la línea 110 se almacena en la variable T el valor de la variable de tiempo (TI) para determinar después de la ejecución el tiempo invertido.

En la línea 120 se inicia un bucle para la ejecución de la subrutina 100 veces (en realidad son 200 veces, como veremos más adelante).

A continuación, en las líneas 130 y 140 se da a las variables X e Y valores aleatorios entre 1 y 25 para X (línea) y entre 1 y 40 para Y (columna). Por su parte en la línea 150 se realiza la llamada a la subrutina propiamente dicha (GOSUB 10) y luego el PRINT del carácter (en este caso "*"). Como se verá más adelante la técnica que utilizamos consiste en situar el cursor en el lugar que deseamos (de lo que se encargarán las subrutinas que estudiaremos) y luego realizar el PRINT (primero apuntar y después disparar).

En la línea 155 nos aprovechamos de la subrutina que estamos estudiando para sacar por pantalla el número de impresiones que se han realizado. Y la línea 160 termina la ejecución del bucle.

En las líneas 200-250 se realiza la impresión final del programa, dando los resultados de ocupación de memoria y de tiempo empleado en la ejecución.

Y ahora veamos las rutinas propiamente dichas. En las líneas 200, 220, 230 y 240 se incluyen entre paréntesis el número

OFERTA DE TRABAJO

Se necesitan colaboradores para traducción de manuales y programas para COMMODORE-64. Imprescindible residir en Barcelona y ser usuario de este equipo.

Interesados llamar a Sr. Pardo Tel.: 321 96 36 - 321 97 58 en horas de oficina

de espacios (xxSPC) necesarios para realizar una presentación correcta del mensaje.

La primera de ellas se desarrolla en los siguientes pasos:

—Poner el cursor en HOME.
—Hacer un PRINT de X-1 caracteres CURSOR ABAJO.
—Hacer un PRINT de Y-1 caracteres CURSOR DERECHA.
—Hacer un PRINT del carácter.

A continuación vemos una subrutina para hacer esto:

```
10 PRINTCHR$(19):
12 IFX<2THEN16
14 FORZ=1TOX-1:PRINTCHR$(17):
   :NEXTZ
16 IFY<2THEN20
18 FORZ=1TOY-1:PRINTCHR$(29):
   :NEXTZ
20 RETURN
```

Que colocada en el programa de evaluación de subrutinas nos ha dado (en un COMMODORE 64) los siguientes resultados:

TIEMPO=39 SEGUNDOS
OCUPACION DE MEMORIA=580 Bytes antes de la ejecución y 622 después de la ejecución.

Esta rutina no tiene ningún secreto exceptuando los IF-THEN de las líneas 12 y 16 que permiten trabajar en la primera línea y la primera columna de la pantalla.

Aunque no es muy rápida ya es un comienzo. Y podríamos ganar tiempo colocando los movimientos de cursor en dos cadenas de caracteres, para ello necesitaremos algún tipo de inicialización al principio del programa:

```
0 GOTO2
2 X$="":FORI=1TO24:X$=X$+CHR$
  (17):NEXTI
3 Y$="":FORI=1TO39:Y$=Y$+CHR$
  (29):NEXTI
```

La línea 0 se modifica para realizar la inicialización en las líneas 2 y 3 que a su vez preparan dos cadenas de caracteres con los movimientos de cursor necesarios. Así la subrutina propiamente dicha quedaría reducida a:

```
10 PRINTCHR$(19);MID$(X$,1,X-1):
   MID$(Y$,1,Y-1):
20 RETURN
```

El cambio en tiempo de ejecución es grande, pues pasamos de 39 a 11 segundos pero el programa antes y después de ejecutarse ocupa 689 y 717 Bytes respectivamente (cuando con la rutina anterior ocupaba 580 y 622). Lo que ocurre es algo muy típico en el uso de ordenadores: si ganamos velocidad perdemos memoria y viceversa.

¿Esta ley fundamental puede ser burlada? La respuesta es SI. Para ello razonemos un poco:

La primera alternativa que se nos ocurre es la de escribir la rutina de posicionamiento del cursor en código máquina lo cual nos permitirá trabajar a la máxima velocidad que permite nuestro equipo, pero resultará complicado incorporar esta rutina en nuestros programas en BASIC y además continuaremos sin resolver el problema de ocupación de memoria. La solución ideal consiste en utilizar alguna rutina del sistema operativo de nuestro ordenador que realice esta función. En el caso del COMMODORE 64 las líneas 10-20 deben ser:

```
10 POKE211,X-1:POKE214,Y-1:
   SYS58732
20 RETURN
```

Con POKE211,X-1:POKE214,Y-1 preparamos la columna y fila en las que queremos trabajar y el SYS sitúa el cursor en el lugar preseleccionado. Los resultados son muy buenos: 8 segundos y 532-567 Bytes. Por último para los equipos con BASIC 4.0 (4032 y 8032) la línea 10 sería:

```
10 POKE 216,X-1:POKE 198,Y-1:
   SYS57447.
```





Rincón del 700

SOFTWARE PARA EL 700 (6): UTILITARIOS (2)

Unit Change y Header Change

Por Jordi Sastre

Le toca el turno a dos utilitarios bastante sencillos y concretos. Uno para cambiar el número de periférico a la unidad de discos (UNIT CHANGE), y otro para cambiar el nombre a un disco (HEADER CHANGE).

UNIT CHANGE

A una CPU de Commodore, no sólo de la serie 700 sino también de las anteriores (8000, 4000, ...), se le pueden conectar varias unidades de disco. El bus IEEE permite formar una cadena de periféricos en principio ilimitada con la única condición de que cada periférico tenga un número diferente, a efectos de una fácil identificación. Por ejemplo, una CPU puede disponer de dos impresoras (una con el número 4 y otra con el 5) y tres unidades de disco (numeradas como 8, 9 y 10), todo ello trabajando interactivamente.

Como todo, los números de periférico también tienen sus reglas. Si enviamos datos hacia el periférico número 4, el ordenador ya sabe que se trata de una impresora, de la misma manera que el 8 corresponde al disco y el 3 a la pantalla. Concretamente, los números de periférico permitidos en el 700 son:

0 - Teclado.
2 - Interface RS 232 C.
3 - Pantalla.
4-7 - Impresoras.
8-15 - Unidades de Disco.

Las unidades de disco son el único periférico al que se le puede cambiar el número sin hacer modificaciones internas de hardware. Es decir, por programa podemos cambiar su número, que permanecerá activo hasta que se apague la unidad o se vuelva a cambiar de número.

La operación en basic es bastante sencilla:

```
OPEN 15,PA,15
PRINT#15,"M-W"CHR$(12)CHR$(80)
 CHR$(2)CHR$(32+PN)CHR$(64+PN)
CLOSE15
```

Siendo PA el número de periférico actual y PN el nuevo número.

La principal ventaja de que esta operación esté integrada en un utilitario es evitar el tener que acordarse de la secuencia de caracteres que han de enviarse al disco.

Esta función actúa sobre cualquiera de las unidades de discos Commodore anteriores y actuales en el mercado: 2031, 2040, 3040, 4040, 8050, 8250, 9060 y 9090. No funciona para las unidades 1540 ni 1541 de ordenadores personales.

La unidad vuelve a su número original en cualquiera de los siguientes casos:

—Desconectando la unidad de discos.

—Desconectando la CPU estando a ella conectada la unidad de discos (por el bus IEEE, claro).

—Reseteando la unidad de discos: OPEN,8,15,"U:":CLOSE15

El manejo del utilitario UNIT CHANGE no tiene ningún problema. Basta con cargar el programa (DLOAD "UNIT CHANGE") y ejecutarlo (RUN).

Una vez en marcha pide el número actual del periférico y el nuevo número. Comprueba que ambos números sean correctos: que estén comprendidos entre 8 y 15 (ambos inclusive), y que el número de periférico actual esté realmente conectado a la CPU.

Si estos requisitos se cumplen, el programa cambia el número y finaliza.

HEADER CHANGE

Cambiar el nombre a un disco puede parecerle una tontería a quien nunca lo haya hecho. Si cambiamos el contenido de un disco, lo normal es que lo volvamos a formatar, asignando entonces el nuevo nombre. Pero por aquello que pasa y nunca se sabe por qué, más de una vez nos vemos en la necesidad de cambiarle el nombre a un disco. De todas maneras siempre es interesante saber que es una cosa que puede hacerse en cualquier momento.

El nombre del disco está almacenado en el bloque de cabecera de directorio. Este bloque se halla en diferente track y sector según el modelo de unidad de discos de que se trate. En la Tabla-1 puede apreciarse la ubicación del nombre del disco en cada modelo:

El espacio destinado a guardar el nombre del disco es siempre de 16 bytes. En caso de que el nombre fuera menor, deberá rellenarse con CHR$(160) hasta completar los 16 bytes.

Hay varias maneras de cambiar el nombre a un disco. Puede hacerse con el utilitario HEADER CHANGE, del que ahora me estoy ocupando. También con el DISKMON, pues gracias a la Tabla-1 sabemos exactamente dónde está ubicado el nombre. O puede hacerse en Basic.

Veamos el siguiente programa:

```
10 INPUT "NUEVO NOMBRE DEL
   DISCO";ND$
20 IF LEN(ND$)<16 THEN ND$=ND$
   +CHR$(160) : GOTO20
30 ND$=LEFT$(ND$,16)
40 OPEN 15,8,15 : OPEN 2,8,2,"#"
50 PRINT#15,"U1";2;0;39;0
60 PRINT#15,"B-P";2;6 : PRINT#2,ND$;
70 PRINT#15,"U2";2;0;39;0
80 CLOSE2 : CLOSE15
```

No es difícil analizar lo que hace. En la línea 10 pide el nuevo nombre que se desea dar al disco. En la línea 20 lo complementa con CHR$(160) en el caso de que se haya introducido un nombre menor de 16 caracteres, y en la línea 30 lo corta por si se ha introducido un nombre demasiado largo.

En la línea 40 abre el canal 15 de comandos y el fichero 2 de acceso directo. En la 50 carga en memoria el bloque correspondiente al track 39, sector 0, pues estamos usando la unidad de discos 8050 ó 8250 (consultar Tabla-1). En la línea 60 escribe el nuevo nombre del disco a partir del byte 6, y en la 70 graba de nuevo el bloque en su lugar.

El utilitario HEADER CHANGE hace exactamente lo mismo que nuestro programa Basic, aunque con las ventajas que un utilitario bien hecho conlleva: posibilidad de modificar el nombre sin tener que teclearlo todo entero, gestión sobre el track y sector pertinentes según el modelo que se utiliza, y posibilidad de cambiar también el identificador del disco.

El funcionamiento del HEADER CHANGE es el siguiente:

En primer lugar pide el modelo de unidad (8050, 8250, 9060 ó 9090), el número de periférico y el drive. Comprueba que estos datos sean correctos. Si es así visualiza el nombre e identificador actuales del disco permitiendo su modificación. Una vez el operador ha validado estos datos, se graban en el disco y finaliza el programa.

MODELOS	TRACK	SECTOR	BYTES
2031 y 4040	**18**	**0**	**144-159**
8050 y 8250	**39**	**0**	**6-21**
9060 y 9090	**76**	**20**	**6-21**

Galería de Soft



Continuación del listado de programas de Microelectrónica y Control

PROGRAMAS VIC-20
Programas en cinta

Programas Educativos

Hangmath 1.500

Sumas y restas para chicos muy jóvenes saliendo naves espaciales para motivarlos. Una manera de resultado final), procurando no ser ahorcado. Las instrucciones son en castellano.

Skymath 1.500

Sumas y restas para chicos muy jóvenes saliendo naves espaciales para motivarlos. Una mara de aprender jugando. 3K.

Space Division 1.500

20 preguntas sobre divisiones muy sencillas. En función de los resultados correctos, despega un cohete que llegará más o menos lejos. 3K.

English Languaje 2.000

Especialmente adecuado para las personas que poseedoras de un buen nivel de inglés, quieren alcanzar cotas superiores. Necesita ampliación de 8K. Presentado en caja y acompañado de su correspondiente manual.

Mastermind 3.000

Participe en el famoso concurso de la BBC, de preguntas y respuestas sobre temas variados. Este cassete va acompañado del sistema operativo y de un file de demostración. Necesita ampliación de 8K. Presentado en caja.

Matemáticas I (Nivel BUP) 2.000

Contiene sistema operativo y 7 cuestionarios sobre diferentes temas abarcando geometría, aritmética, matemáticas generales, álgebra, etc. Necesita ampliación de 8K ó 16K. Presentado en caja.

Programas aplicaciones

Agenda telefónica 10.000

Programa que permite llevar una agenda telefónica. Tiene una capacidad de 300 números de teléfono. Presentado en disco. Manual en castellano.

PROGRAMAS COMMODORE 64

Programas Educativos

Music Machine 4.500

Ejercítese tocando un teclado musical, con acompañamientos. Presentado en cartucho.

Music Composer 4.500

Este cartucho le permite componer su melodía, favorita y guardarla en un cassette. Presentado en cartucho.

Visible Solar System 4.500

Cartucho educativo en el cual usted tendrá imágenes de los planetas y las órbitas que éstos describen en el sistema solar. Presentado en cartucho.

Speed/Bingo Math 4.500

Son dos tipos de juegos. En cada uno le aparece una pantalla sobre la cual usted debe resolver los problemas planteados. Presentado en cartucho.

Física I 3.500

Este disco contiene varios programas para el uso educativo en el área de FISICA. Estos programas tocan como: Funcionamiento del péndulo simple; Electricidad, estudios sobre resistencias en serie y en paralelo; Atomo de Bohr; El principio de Arquímides. Presentado en disco. Programas en catellano.



5 cintas para divertirse

Están fabricadas en Inglaterra, por las prestigiosas firmas Imagine Software y Technology Group Ltd, y las distribuye en España por correo Micro Informática Soft, S.A. (M.I.S.S.A.), con explicaciones e instrucciones completas en español. Estos cinco programas los hemos estado probando en nuestra redacción. Los hemos encontrado muy entretenidos, bien confeccionados y de muy buen funcionamiento, por lo que publicamos a continuación una reseña resumen de su contenido.

Pedro

(2.325 Ptas.) C-64

No hay tiempo para que Pedro se eche la siesta, es el Jardinero más ocupado de México.

Le ha costado muchos años de trabajo y de mimos para llegar a tener el jardín de flores más bonito del país. El problema es que todos los animales del barrio han adquirido un gusto para las flores suculentas de Pedro y siempre están buscando la forma de entrar en el jardín para comérselas.

Pedro trabaja de sol a sol y los animales no le dejan ni un momento de paz. Intenta por todos los medios evitar que los animales coman sus flores, desde bloquear los senderos del jardín con abono y ladrillos, hasta ahuyentar a los animales o saltar encima de ellos: Pero con todo, algunos llegan a colarse.

Cada vez que un animal consigue entrar y comerse unas flores, Pedro tiene que sembrar más. Un problema viene detrás de otro, y, por si fuera poco, el vagabundo del barrio entra de vez en cuando e intenta robar unas semillas y Pedro tiene que estar pendiente de él también.

Entre los animales intrusos y el vagabundo ladrón, parece que tu tampoco vas a tener tiempo para echarte la siesta.

Así que, animate, esto no se puede dejar para mañana.



FINALIDAD DEL JUEGO

Para evitar que los animales se coman las plantas, Pedro tiene que recoger los ladrillos o el abono de las esquinas inferiores de la pantalla para bloquear las salidas del laberinto. No puede llevar o colocar más de un objeto a la vez.

Cualquier animal que logre entrar en el jardín se dirigirá hacia las plantas y empezará a comérselas. Tu tienes que decidir si prefieres bloquear el laberinto o encargarte de los animales, y luego tienes que sembrar las calvas en el jardín, en cuanto te sea posible.

Arcadia-64

(2.150 Ptas.) C-64

Tu eres el capitán de la nave estrella ARCADIA. La más sofisticada tecnologia especial jamás inventada está bajo tu control, incluyendo el impresionante motor Ion Thrust, y la energía imponente del arma doble Plasma Disruptor. Con todo esto, tú eres el ser más potente de la galaxia... y, para algunas personas, el más odiado.

El pueblo atariano ha ido ampliando su imperio, a base de esclavizar poco a poco a los planetas más pequeños y más vulnerables. Ahora mismo se encuentra en una inmejorable situación, y está dispuesto a esclavizar la galaxia entera.

La única fuerza capaz de rechazar el ataque de las hordas de Atari está en tus manos: la nave estrella ARCADIA. Cada fibra de tu cuerpo tiembla de tensión mientras te preparas para enfrentarte al ataque de la flota de guerra de Atari.

Atacan al Arcadia en oleadas, cada ataque con un tiempo de duración fija. Si consigues destruir la flota entera dentro de este tiempo fijo, otra flota te atacará. Si alguna raza fracasa en el intento de destruir el Arcadia, otra raza continuará el ataque.

Cada vez que se inicia un ataque nuevo, este consistirá en unas naves más sofisticadas y más mortíferas que las anteriores. ¿Puedes seguir defendiéndote contra el imperio de Atari? La suerte de toda la galaxia está en tus manos, ¿tú tiénes la fuerza de voluntad y la habilidad para salir adelante? ¿Eres capaz de salvar la civilización?

La cinta contiene 8 variaciones posibles de juego.

El Camarero Enloquecido

(Wacky Waiter)

(1.725 Ptas.) VIC-20

Tu trabajo como asesor del gobierno para el ocio no te convence mucho, así que decides buscar el método de ganarte unas "pelas" extras. Llevas tiempo pateando las calles sin haber encontrado nada, hasta que de repente tropiezas con un cartel ofreciendo trabajo en un pequeño hotel metido en un callejón tranquilo. Al entrar en este lugar destartalado descubres que el trabajo es de camarero, y tu situación desesperada te obliga a solicitar el empleo. El jefe de aspecto mezquino decide aceptarte, pero te avisa que no te quitará el ojo de encima.

Pronto te ves ataviado en un chaleco y pantalón elegantes y dispuesto a atender cualquier capricho de los clientes. Te encuentras en la recepción esperando tu primera tarea, que no tarda en llegar. Un huesped empieza a pedir servicio a gritos, así que tú echas a

correr para ver lo que quiere. Tienes que ir a la tercera planta, así que coges el primer ascensor que pasa y bajas en dicha planta. Pero el cliente se encuentra al otro extremo y tu sólo puedes llegar a él mediante los ascensores situados en medio. Tienes que esperar que pase el siguiente ascensor pero lo pierdes. Mientras tanto, tu propina se va reduciendo, pero ya aprenderás. El ascensor pasa por fin y lo coges corriendo. Ahora que has cogido el truco, sigues cogiendo ascensores hasta que llegues a la planta donde tienes que bajar. Allí se encuentra el cliente pidiéntote ¡¡VINO!! a gritos. Te diriges a la planta donde sirven el vino para recogerlo y llevarlo al cliente y aceptar la pequeña propina que te ofrecen. Sólo te pagan en propinas, así que más vale que te vayas cogiendo el truco rápidamente. Vuelves a la recepción donde te espera otra llamada y sales corriendo de nuevo. Pero acuérdate que el jefe siempre te está espiando y si te caes camino al ascensor, recibirás un aviso por escrito, pero si él te ve, te despedirá en seguida. La cantidad de la propina continúa reduciéndose, y si no llegas a servir al cliente antes de que la propina llegue a cero, el cliente beberá su



copa de un trago y, totalmente borracho, saldrá en tu búsqueda. Si te encuentra, no estará muy contento de verte y te tirará las copas por todos lados, provocando otro aviso por escrito de parte del jefe. Por ahora, ya van dos avisos, y uno más y te encuentras en la calle. Así que ojo.

Efectivamente, la vida de camarero no es nada fácil.

Hechizado

(Bewitched)

(1.875 Ptas.) VIC-20

La muerte y el hambre dominan la tierra mientras crecen las fuerzas y la astucia de los poderes del mal. En un momento de descuido tuyo el mago malévolo, Mordread, te ha lanzado a las profundidades del laberinto embrujado debajo de su lúgubre castillo.

El mago te ha convertido en una llave mágica destinada a vagar eternamente por los laberintos embrujados escondidos bajo tierra, mientras él esclaviza a todos los que viven en la superficie.

Si no fuese por el hecho de que tú eres el mago más brillante de la tierra estarías condenado para



siempre; pero, a medida que vayas pensando en los siglos de tu vida dedicada a mago, de repente se te ocurre una solución a tu problema: es una solución en teoría, pero ¿serías capaz de llevarlo a cabo?

Existen cuatro juegos de puertas de distintos colores, y hay que abrir cada una de ellas antes de alcanzar la libertad, y mientras tanto te persiguen hordas de fantasmas y demonios que intentan impedir tu progreso.

A la entrada del laberinto se encuentran cuatro llaves coloradas; los colores corresponden a los de las puertas. Tú tienes que pasar a través de una de estas llaves para que te conviertas en el mismo color, y solamente en este caso podrás abrir las puertas correspondientes. puertas correspondientes.

Una vez abiertas todas las puertas del mismo color tienes que volver a la entrada, pasar por otra llave para cambiarte de color, e iniciar el ataque a otro juego de puertas.

Este proceso tiene que repetirse hasta que todas las puertas hayan sido abiertas y te permita alcanzar la libertad: pero cuanto más te vayas acercando al momento de alcanzar la libertad más frenéticas se ponen las hordas de Mordread en sus intentos de impedir tu éxito.

Tienes que escaparte de este lugar húmedo y lúgubre, para que puedas combatir las fuerzas de la oscuridad y traer la luz y la libertad a la tierra de nuevo.

El Super Detective



(Catcha Snatch)

(1.725 Ptas.) VIC-20

Barney Bootlace nunca lo ha pasado tan mal, de ser detective para la aristocracia, ha pasado a ser un humilde detective del almacén peor organizado de la ciudad. Pasea aburrido por los mostradores de joyas y relojes, pensando en su lamentable situación. La vida se presenta difícil para un duro luchador en contra del crimen cuando se pasa el día recogiendo a niños perdidos para llevarlos a la oficina de niños perdidos situada en la esquina superior izquierda del almacén. También tiene que llevar paraguas y bolsos extraviados al despacho de objetos perdidos situado en la esquina superior derecha del almacén. La acción se anima un poco cuando algún ratero de poca monta intenta llevarse alguna caja del mostrador. Barney tiene que arrebatarle la caja y devolverla a la oficina del jefe como prueba, pero ojo con el ratero. Mientras tu intentas hacerte con la caja, el ratero la ocultará debajo de su abrigo, y tienes que vigilarle por si se pierde entre la gente. El jefe se encuentra en frente de la salida principal situada en la esquina inferior izquierda del almacén. Hay un momento en que nuestro héroe tiene motivo para perder la calma —cuando un cliente de vez en cuando deja caer una bomba paquete detrás de los mostradores. La única opción para Barney es de coger la bomba y salir corriendo con ella hacia la doble puerta de salida. Por supuesto, si logras realizar esta hazaña, llevas buen camino hacia una promoción inmediata al siguiente nivel. La única meta que tiene Barney en la vida es la de conseguir la promoción al siguiente nivel de juego ganándose el respeto del jefe. Evidentemente, si esta opción queda por debajo del 0%, Barney recibirá un aviso por escrito y será degradado por un nivel. Al recibir tres avisos por escrito, será despedido. Siempre puedes aumentar tus posibilidades subiéndote dos niveles, ya que el jefe te quitará un aviso. Tienes que estar pendiente siempre de la hora



porque a las seis de la tarde hay una avalancha de gente comprando.

Estarás deseando que lleguen las ocho para que cierre el almacén, pero estate preparado para el día siguiente porque llegarán verdaderas hordas de amas de casa que salen de compras.



DESCRIPCION ALFABETICA DE LOS MNEMONICOS DEL 6502/6510 (V)

CMP

Compara memoria con acumulador

Operación: A-M

(Ref.: 4.2.1)

N Z C I D V
√ √ √ — — —

Modo de Direc.	Formato en ensamblador	Código Operan.	Núm. Bytes	Núm. Ciclos
Inmediato	CMP #Oper.	C9	2	2
Pág. Cero	CMP Oper.	C5	2	3
Pág. Cero, X	CMP Oper., X	D5	2	4
Absoluto	CMP Oper.	CD	3	4
Absoluto, X	CMP Oper., X	DD	3	4*
Absoluto, Y	CMP Oper., Y	D9	3	4*
(Indir., X)	CMP (Oper., X)	C1	2	6
(Indir.), Y	CMP (Oper.), Y	D1	2	5*

*** Suma 1 si se cambia de página.**

CPY

Compara memoria con registro Y

Operación: Y-M

(Ref.: 7.9)

N Z C I D V
√ √ √ — — —

Modo de Direc.	Formato en ensamblador	Código Operan.	Núm. Bytes	Núm. Ciclos
Inmediato	CPY #Oper.	C0	2	2
Pág. Cero	CPY Oper.	C4	2	3
Absoluto	CPY Oper.	CC	3	4

CPX

Compara memoria con registro X

Operación: X-M

(Ref.: 7.8)

N Z C I D V
√ √ √ — — —

Modo de Direc.	Formato en ensamblador	Código Operan.	Núm. Bytes	Núm. Ciclos
Inmediato	CPX #Oper.	E0	2	2
Pág. Cero	CPX Oper.	E4	2	3
Absoluto	CPX Oper.	EC	3	4

DEC

Compara memoria con una unidad

Operación: M-1 → M

(Ref.: 10.7)

N Z C I D V
√ √ √ — — —

Modo de Direc.	Formato en ensamblador	Código Operan.	Núm. Bytes	Núm. Ciclos
Pág. Cero	DEC Oper.	C6	2	3
Pág. Cero, X	DEC Oper., X	D6	2	6
Absoluto	DEC Oper.	CE	3	6
Absoluto, X	DEC Oper.	DE	3	7

casa de software s.a.

PARA COMMODORE 64

PRACTICALC 64



MANUAL EN CASTELLANO

PROCESADOR DE TEXTO

35.000 caracteres, 240 columnas
Versión diskette: 21.500,–
Versión cartucho: 24.900,–
(grabación de documentos en diskette y cassette)

HOJA DE CALCULO

2.000 coordenadas
(funciones matemáticas, sort, gráficos...)
Versión cassette: 15.500,–
Versión diskette: 17.500,–

VIZAWRITE



MANUAL EN CASTELLANO

CONTABILIDAD 64 PROFESIONAL



300 cuentas
3.000 apuntes por disco
Listado de diario
Balance de sumas y saldos
Balance de situación
Extractos de cuenta
Listado de ficheros
Diario de cierre
Utilitarios, etc...

EQUIPO NECESARIO:
Ordenador: COMMODORE - 64
Unidad de disco: VC 1541
Impresora: VC 1525 ó MPS 801
Monitor ó T.V.

P.V.P. 24.550,–

ADAPTADO AL PLAN GENERAL CONTABLE ESPAÑOL

CONTABILIDAD DOMESTICA

1 cuenta ingresos
15 cuentas gastos
Listados por impresora y pantalla.
Análisis porcentual
Gráfico de gastos por pantalla.

VERSION CARTUCHO

P.V.P. 14.500,–

LAPIZ OPTICO Y SOFTWARE GRAFICO



Dibujar en pantalla a mano alzada formas geométricas, sombreados... Grabación del dibujo en cassette.

P.V.P. 12.500,–

INTER FACE IEEE 488

Para conectar cualquier periférico con protocolo IEEE 488 (floppys 8050, 8250, impresoras 8023, 8024 etc.) al COMMODORE 64



P.V.P. 29.000,–

JOY STICK DE PRECISION



Pulsador sensible
Cable extra largo
Para CBM 64, VIC 20, ATARI CX 2600

P.V.P. 2.950,–

CONTABILIDAD DOMESTICA

22 cuentas
Listado por pantalla o impresora de los ingresos y gastos de cada período definido.

VERSION CASSETTE para
C-64: 3.500,–
VIC-20: 2.500,–

BOLETIN DE PEDIDO

Nombre y dirección:

☐ Contabilidad 64 (profesional)
☐ Contabilidad doméstica (cartucho)
☐ Contabilidad doméstica (cassette-C 64)
☐ Contabilidad doméstica (cassette-VIC 20)
☐ Practicalc ☐ Vizawrite
☐ Joystick ☐ Lápiz óptico
☐ Interface IEEE 488
☐ Información detallada

FORMA DE PAGO

☐ Adjunto talón (añadir 250 pts. por producto para gastos de envío)
☐ Contra reembolso

Enviar a: **CASA DE SOFTWARE S.A. Aragón, 272, 8.º 6.º Barcelona-7** tel. 215 69 52

Magia



La MAGIA son trucos, la MAGIA es divertida.

La MAGIA es hacer lo que nadie se ha atrevido.

La MAGIA es una columna mensual llena de consejos, trucos, de esto y aquello del mundo del software, hardware y aplicaciones.

Cada mes, MAGIA les trae trucos breves y útiles de informática procedentes de todo el mundo - trucos descubiertos por los demás que hacen que la informática sea más fácil, más divertida o más animada.

MAGIA habla de ideas sencillas, programas de una sola línea, subrutinas útiles, hechos de informática poco conocidos y otras cosas de interés. Buscamos material nuevo o renovado que resulta ser de valor actual para usuarios de equipos Commodore y que puede utilizarse con un mínimo de tiempo, esfuerzo o conocimientos teóricos.

MAGIA resulta ser la fuente más completa de información para la informática práctica, además de ser un foro internacional para compartir trucos con otros aficionados. Envíe sus trucos a:

COMMODORE WORLD
Pedro Muguruza, 4
Madrid-16

La revista "Commodore World" sorteará seis paquetes de software en julio y diciembre entre todas las contribuciones publicadas.

Probé una de las "recetas mágicas" publicadas en el nº 1, la de la colocación en el centro de la pantalla del VIC. En primer lugar, según indica, las direcciones de memoria son iguales para el control horizontal y vertical (36864) de la pantalla. Por otra parte si hacemos un poke con el valor 5, la pantalla se desplaza a la izquierda, y si lo hacemos con 25, hacia la derecha. Es decir, esta posición sólo controla el centro horizontal de la pantalla. Los valores correctos son: 36897 que controla la posición vertical y 36896 que controla la horizontal. Para ajustar la pantalla sólo es necesario incluir estas direcciones de tal forma, en un programa, que lea el peek inicial y este aumente o disminuya según una tecla que se pulse, con lo que controlamos la posición hacia arriba, abajo, izquierda o derecha según la tecla pulsada.

Mario Díaz Gañán
Cte. Zorita, 25
Madrid-20

PRINT USING

Los que echen de menos la existencia del comando PRINT USING en el 64, pueden suplirlo en parte utilizando:

```
DEF FNU(X)=INT(X*1Ø ↑ D+.5)/1Ø ↑ D
```

al principio del programa.

Cuando se quiera imprimir A con 2 decimales, basta la siguiente sentencia:

```
1ØØ D=2:PRINT FNU(A)
```

Guillermo Camaño Olivares
Arabial, 11-1º D
Granada-3

Hola amigos de **Commodore World.**

Me llamo Miguel Luis Izquierdo y tengo 13 años.

Tengo el gusto de mandaros esta carta para presentaros un par de curiosidades Pokes para el VIC-20.

El primero hace de las teclas del VIC, teclas **repetitivas**.

POKE36987, 121.

Y el segundo mucho más útil es capaz de controlar la velocidad del cursor, cuanto mayor sea la cifra de después de la coma, más lento se desplazará y si ponéis una cifra muy baja como 12 ó hasta 1 podrá coger velocidades de espacio.

También es muy útil para juegos ya que poniéndolo en la modalidad 1 aquellos juegos que no lográis dominar los tendréis a "cámara lenta" y los juegos que sean anticopias tendréis que cargarlos y sin necesidad de añadirlos al programa ni ponerles número de línea, bastará.

POKE36983,1 para juegos y según cambiéis la cifra para control del cursor.

Mi teléfono es el 729 39 82, San Modesto, 46. Madrid.

P.D.: Soy el hijo de José Antonio Izquierdo, socio del Club Commodore World.

Miguel Luis, publicamos tus trucos en esta sección, pero siento tener que decirte que el primero de los dos sólo funciona algunas veces, el POKE correcto para hacer repetitivas todas las teclas es POKE650,255. Respecto al segundo POKE es correcto y funciona muy bien, sigue enviando todos los trucos que descubras y los publicaremos.



AUTO-RUN

Poniendo estas dos líneas al final de un programa, por ejemplo un cargador de caracteres propios, hacen que se cargue automáticamente el programa que sigue en la cinta y lo empiece a rodar sin necesidad de hacer 10AD y RUN.

```
5000 POKE 198, 7: POKE 631,76: POKE632,207:
     POKE633,159:POKE634,13
5010 POKE 635,82: POKE636,213:POKE637,13
```

Angel León Negrín
Edificio Ahemon Porton 3 Bajo A
Polígono de Ofra
Santa Cruz de Tenerife

EL VIC PATRIOTA

Soy un suscriptor de la revista Commodore World, y quiero aportar mi granito de arena enviándoles el **programa de una línea** en la sección de MAGIA por si les parece bien publicarlo:

```
1PRINT(CLR):POKE 36879,41:FOR A =4228 TO 4469:POKE
A,102:POKE33792+A,7:NEXT:FOR B=1TO2↑99:NEXT
```

Está hecho en un VIC-20 con más de 8 K; si tuviéramos menos, sólo hay que sustituir los números :4228,4469 y 33792 por 7812,8053 y 34304 respectivamente su longitud es de 66 bytes y construye la Bandera española.

Fdo. Federico Iborra Abargues
C/ Comandante Zorita, 8
Madrid-20

NADA POR AQUI... NADA POR ALLA... Y AQUI APARECE LA LINEA

Pulsando la tecla (shift) o (comm.) junto con la tecla inst/del, así durante un poco, soltaréis la tecla shift y veréis cómo aparece una en inverso de color. Colocando este signo tantas veces como caracteres haya en la línea, contando el número de líneas conseguiréis que esta línea desaparezca del listado. Deberéis colocarlo con un Rem, abriendo comillas pero no cerrándolas, porque si no estas comillas aparecerán en el listado.

La forma de poder destruir ésta te será aplicando este pequeño programa pero sin número de líneas.

```
FOR I=4096 TO 7679:PRINT I, CHR$(PEEK(I)):NEXT I
```

Cuando os encontréis unas comillas con un corchete en inversor de color, rectifico varios corchetes en inverso. Puede que haya detectado un PRINT o un REM anteriormente nombrado. Si creéis que es un REM hacer un POKE con el número que había a su izquierda, es decir POKE número, 0. Si era un REM veréis lo que había escondido, pero si era un PRINT perderéis lo que había entre comillas. Se me olvidaba para descubrirlo veréis un parpadeo en la pantalla.

VIC-20

Emilio Rocafort Riaza
Urbanización Torres San Lamberto
Chalet nº 17 A
Zaragoza

ENCABEZAMIENTO DE LISTADOS

Se puede encabezar un listado añadiendo una primera línea REM, seguida esta sentencia de 6, 7 u 8 Tes invertidas y la frase de encabezamiento. Para obtener las Tes es preciso pulsar comillas, borrarlas, pulsar SHIFT e INST-DEL 6, 7 u 8 veces, volver ante las comillas (SIN BORRARLAS) y pulsar solamente INST-DEL otras 6, 7 u 8 veces. No cerrar las comillas, pues si no aparecerían en el encabezamiento. Ejemplo:

```
10 REM"(7INST-DEL) encabezamiento listado
20 print"(CLR)"
```

Al listar el programa quedaría así:

```
ENCABEZAMIENTO LISTADO
20 PRINT"(CLR)"
```

Carlos López Benavente 14 años.
Pl. Santoña
Edif. Constanza
Murcia-6

DESPROTEGER PROGRAMA

En ocasiones el VIC-20 no refleja en el listado lo que se encuentra en su memoria interna. Por este motivo al correr el programa nos dará error. Cuando ocurre este incidente se vuelve a escribir la línea tal y como está y desaparecerá la nota de error.

Andrés Pozo (Bilbao)
(Mándanos dirección)

¿QUE HORA ES?

He aquí un pequeño programa que puede resultar práctico. Se trata de un modelo de reloj digital con el cual se puede por ejemplo, medir las pulsaciones por minuto.

Cambiando la velocidad del tiempo (T), se puede conseguir un margen de error menos grande, según el tiempo que se desee medir

```
10 PRINT " ♡ "
20 FOR HO=0 TO 24
30 FOR MIN=0 TO 59
40 FOR SEG=0 TO 59
50 PRINT HO,MIN,SEG:FOR T=1 TO 900:NEXT
60 NEXT:NEXT:NEXT:NEXT
```

Siendo ♡ las teclas "SHIFT" y "CLR/HOME")

Con cuatro bucles, insertados, se presenta el esquema de este programa, siendo el mayor para las horas, y los contenidos para minutos y segundos, junto con el más pequeño que controla la velocidad de tiempo.

Por favor, escribir sólo Diego Morales (Huesca)
(Felicidades por la revista, es muy buena).

...Y SEGUIMOS DE HORARIO...

Cuando le preguntamos al VIC 20 la hora, éste nos la expresa de una manera a la que no estamos acostumbrados a ver en los relojes, sin diferenciar horas, minutos y segundos.

En la línea que os expongo a continuación, intercalo dos puntos entre los espacios de horas-minutos y minutos-segundos.

```
100 PRINT "(HOME)"MID$(TI$,1,2)":"MID$(TI$,3,2)":
    "RIGHT$(TI$,2):GOTO 100.
```

Luis Casi Gil (Barcelona)
C/ La Palma, 9 Entlo. 4.ª
Barcelona-4

BLOQUEO DE RUN/STOP Y RESTORE VIC-20

Me llamo José de la Fuente y tengo 16 años. Creo que su revista es fenomenal.

Espero que su revista siga publicándose para que las personas que utilizan ordenadores COMMODORE puedan disfrutar de esta fenomenal revista.

Como piden colaboraciones les mando este pequeño truco para que si creen conveniente lo publiquen.

La línea que doy a continuación, se puede teclear directamente o introducirla al principio del programa.

Se utiliza para el bloqueo de RUN/STOP y RESTORE.

Una vez puesto RUN la única forma de parar el programa es desconectar el aparato.

```
POKE 37150,2:POKE 788,194
```

El primer POKE anula la tecla RESTORE y el segundo POKE anula la tecla RUN/STOP.

José de la Fuente (Barcelona)
Tel. (93) 243 11 41
PO.BOX. 22.204
16 años

NUMEROS PRIMOS

"La rutina indicada a continuación se encarga de seleccionar los números primos comprendidos en el intervalo numérico que se indique desde el teclado. Se debe introducir primero el límite inferior y a continuación el superior."

```
1 PRINT"NOS. PRIMOS DESDE...":INPUT A%:PRINT
  "HASTA...":INPUT B%:IF A%>B% THEN 1
2 FOR I=A% TO B%:IF I=1 OR I=2 THEN 7
```



```
3 C=I  .5:D=INT(C):IF C-D=Ø THEN 8
4 FOR J=2 TO D
5 IF I/J-INT(I/J)=Ø THEN 8
6 NEXT J
7 PRINT I,
8 NEXT I
```

Rafael García Segura
Avda. Barcelona, 40 - 2º
Málaga-9

INSTRUCCION DIRECTA

Para control de juegos es habitual, en lugar de GET, usar el contenido de la posición 197, para detectar qué tecla se está pulsando, en el VIC o en el 64.

Para ello pruebe la siguiente línea, en modo directo:

FOR A=1 TO 1000:PRINT PEEK(197):NEXT

Pulse RETURN y a continuación pulse varias teclas; cada una dará un valor distinto, que puede usar en una condición IF PEEK(197)=##... y tendrá un control continuo y directo del teclado sobre el programa. La línea FOR en modo directo nos permitirá además, observar el contenido de cualquier posición de memoria y su variación, durante unos segundos sin afectar para nada al programa que tengamos introducido en este momento.

Francesc A. Castanyer P.
C/ Queretaro, 5, E-2ª, 2º D
Palma de Mallorca, 7
Baleares

Queridos conVICcinos:

Os mando un truqui para la sección magia, no sé si ya lo sabe todo el mundo y estoy pasándome de listo pero si os parece que no vale la pena pues al cubo de archivo.

Cuando se redefine en RAM el generador de caracteres con caracteres propios, etc., se pierde en caso de listar el centelleo del cursor siendo muy complicado el distinguir por donde anda cuando está entre las líneas del listado, el truco es cambiar su color a negro por ejemplo con (CTRL-BLACK) siempre que el color de pantalla no sea negro.

José Luis Lozano
Rambla Justo Oliveras, 81, 4º 1
Hospitalet
(Barcelona)

COSA DE MAGIA

En un alarde de virtuosismo, sitúe la revista a modo de partitura frente al VIC, enciéndalo y teclee:

POKE 56,PEEK(44):SYS 58234 [RETURN]

No se responde de FRE(0)

Para volver a la realidad use SYS 64802 [RETURN] o apague y váyase.

Si aún está ahí, apague, encienda de nuevo, y a continuación ponga lo siguiente:

POKE 43,0:POKE 56,PEEK(44):SYS 58234 [RETURN]

Un consejo: No se crea nada de lo que vea en pantalla.

Francesc Castanyer P.
C/ Queretano, 5, E-2.ª, 2 D
Palma de Mallorca, 7
Baleares

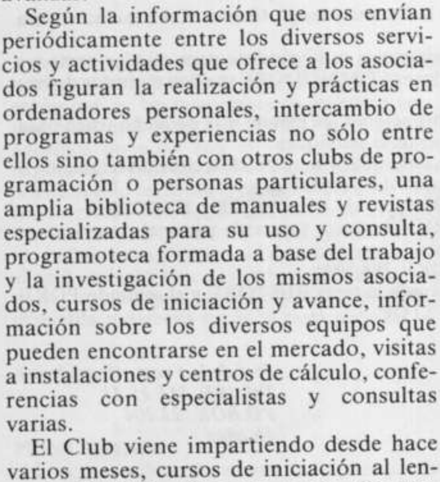
NOTICIAS COMMODORE WORLD



Club Alaiz

El Club Alaiz de nuevo con nosotros

El Club Alaiz no deja de moverse y avanzar.



Según la información que nos envían periódicamente entre los diversos servicios y actividades que ofrece a los asociados figuran la realización y prácticas en ordenadores personales, intercambio de programas y experiencias no sólo entre ellos sino también con otros clubs de programación o personas particulares, una amplia biblioteca de manuales y revistas especializadas para su uso y consulta, programoteca formada a base del trabajo y la investigación de los mismos asociados, cursos de iniciación y avance, información sobre los diversos equipos que pueden encontrarse en el mercado, visitas a instalaciones y centros de cálculo, conferencias con especialistas y consultas varias.

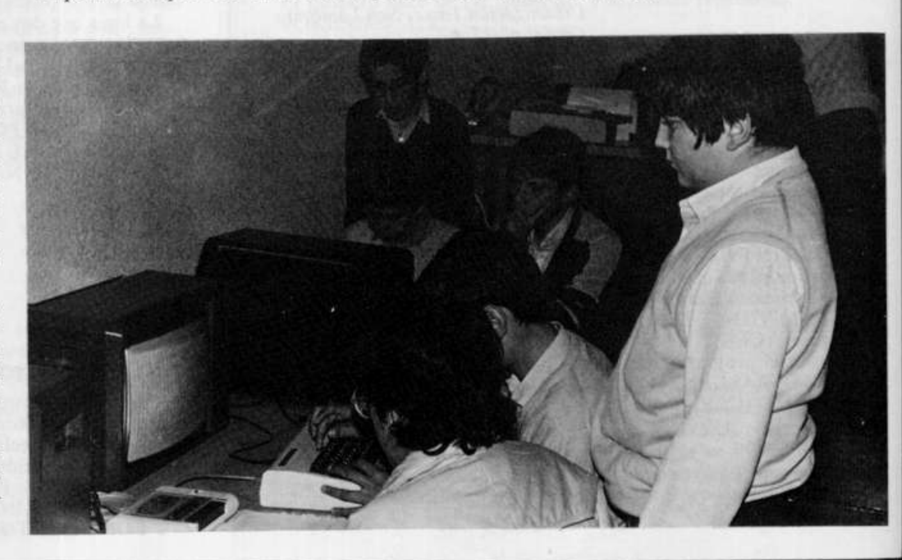
El Club viene impartiendo desde hace varios meses, cursos de iniciación al lenguaje BASIC utilizando para ello tres equipos VIC-20 y un COMMODORE-64 con diversos periféricos, gracias a la colaboración de Comercial Noain.

Como resulta lógico tratándose de un Club de reciente creación, se encuentra abierto a todo tipo de ayudas o intercambios de experiencias y programas.

Nos dicen que han elaborado ya algunos programas propios originales, y que aunque está dirigido principalmente a estudiantes de BUP y COU, van a comenzar próximamente unos cursos de iniciación para profesionales, base para otras posibles extensiones en el futuro.

Esperamos que nos envíen bien pronto algunos de estos estupendos programas a fin de poder compartir con el gran Club Nacional Commodoriano de todos nuestros lectores sus experiencias.

La sede del Club de Programación Alaiz está sita en la plaza Monasterio Santa Gemma s/n. Pamplona. Teléfonos: 25 44 80/25 77 04.

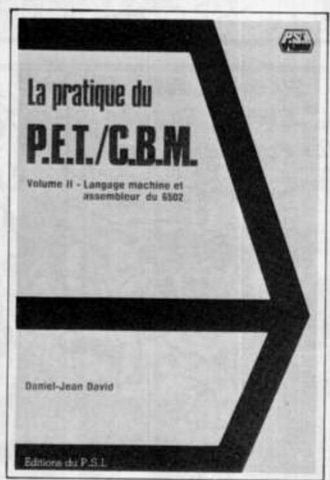




DE TOMO Y LOMO

Según prometimos en nuestro último número hemos iniciado la búsqueda de información sobre los ordenadores PETS. La literatura existentes sobre este tema es muy escasa, pero nuestros héroes han intentado por todos los medios cumplir con su misión. Lo único que os podemos ofrecer de momento son cuatro publicaciones, y además en francés!!! No os podemos presentar aquí un análisis detallado de estos libros por no haber tenido el tiempo suficiente para estudiarlos a fondo, pero aparecerá en el próximo número. Los distribuye P.S.I. Ibérica, Ferraz, 11-3º, Madrid, 8, teléfono: (91) 247 30 00. Los precios y los títulos son: "Clefs pour le P.E.T./C.B.M." = 1.700 peetas, "La Pratique du P.E.T./C.B.M. Volume 1- Perpheriques et Gestion des Fichiers" = 1.450 pesetas, "La Pratique du P.E.T./C.B.M. Volume 11 - Langage machine et assembleur du 6502" = 1.700 pesetas, "La decouverte du P.E.T./C.B.M." = 1.450 pesetas. El autor de los cuatro libros es Daniel-Jean David.









MICRO INFORMATICA Y SOFT, S.A. (M.I.S.S.A.)
PRESENTA
5 JUEGOS 5

ACTUAN:

PEDRO: (Cuando no tiene que proteger sus flores de los ataques de los animales y los vagabundos.)

EL SUPER DETECTIVE: (Sólo los domingos, por ser el día en que no trabaja en el almacén ocupándose de los cacos, los niños perdidos, y las bombas.)

EL CAMARERO LOCO: (Sólo en días de fiesta, ya que los demás se dedica a no perder su empleo en el hotel peor organizado del mundo.)

HECHIZADO: (Si consigues "deshechizarte" a lo mejor tú también.)

ARCADIA-64: (Con la nave estrella más potente bajo tu control puedes hacer lo que se te antoje, si no te vence Mordread, claro.)

COPYRIGHT: IMAGINE SOFTWARE LIMITED.
(ESTE CARTEL NO ES PARA TURISTAS: AQUI DECIMOS TODO EN ESPAÑOL)
VENTA SOLAMENTE POR CORREO

BOLETIN DE PEDIDO

INCLUYO CHEQUE NOMINATIVO (M.I.S.S.A.) POR .... pesetas.

PAGARE CONTRA REEMBOLSO LA CANTIDAD DE .... pesetas.

MAS 150 pesetas de gastos de envío (de 1 a 5 cassettes).

ENVIAR A: M.I.S.S.A. MAESTRO CHAPI, 15. MADRID-16

NOMBRE ....
CALLE .... Nº ....
POBLACION .... DP ....
PROVINCIA .... TEL. ....

DESEO RECIBIR LOS SIGUIENTES PROGRAMAS:
EL CAMARERO LOCO (VIC-20) .... Precio 1.725 ... ☐
EL SUPER DETECTIVE (VIC-20) .... Precio 1.725 ... ☐
HECHIZADO (VIC-20) .... Precio 1.825 ... ☐
ARCADIA-64 (C-64) .... Precio 2.125 ... ☐
PEDRO (C-64) .... Precio 2.350 ... ☐


Las Aventuras de Ramy Romo y sus amigos
¡Hola! Ramy Romo
por Errazquin
UNOS INTRUSOS... HE DE AVISAR INMEDIATAMENTE AL EMPERADOR DE LA LOGICA...
BZZZZZ
VOLARE A MI MAXIMA VELOCIDAD
ZIUUUUUU
POM
EHH! ESA COSA NOS ESTABA ESPIANDO


¿?
COMMODORE WORLD
ZZZZ
ZIUUUU
BONK


ME IMAGINO QUE TENDRAS UNA BUENA EXCUSA PARA INTERRUMPIRME ASI....
SI... AMO
COMMODORE WORLD


GFF... HAY... ¡GLUP! .. DOS NUEVOS COMPONENTES EN EL ORDENADOR... AY.. QUE GOLPEEE..


¡ESTA BIEN.. BUG! PIENSA... CON LOGICA. DIME CUALES SON LOS COMPONENTES QUE HAS VISTO.. HUMMM
COMPONENTES
CHIP
TECLA
CONECTOR
11001001
BYTE
HZ
PANTALLA
RELOJ
MANGA
PUES.. PUES.. ES QUE.. ¡GLUP! NO ESTAN EN ESE MAPA...


¿COMOOO? NADIE PUEDE PONER UN COMPONENTE NUEVO SIN MI REAL E IMPERIAL PERMISOOO


¡LARGOOOO! ¡CAPTURADME A ESOS DOS INMEDIATAMENTE.. FUERAAAA!
POM


TU Y TUS TONTERIAS.. PUES YO A CASA DE MR. CHIP NO VOY, QUE TODAVIA ME DUELEN LOS CABLES DE LA PALIZA...


1001
1100


HUM.. ¡VEAMOS QUIEN CANTA EN BINARIO.. .. NO SON LOS QUE BUSCO.. PERO SERVIRAN PARA CALMAR EL CABREO DEL EMPERADOR.!


¡ATRAPEMOSLOS!
..¿?

HE DE AVISAR A MR. CHIP DEL RAPTO..
EEEH ¡ESE NOS HA VISTO!
SSSS

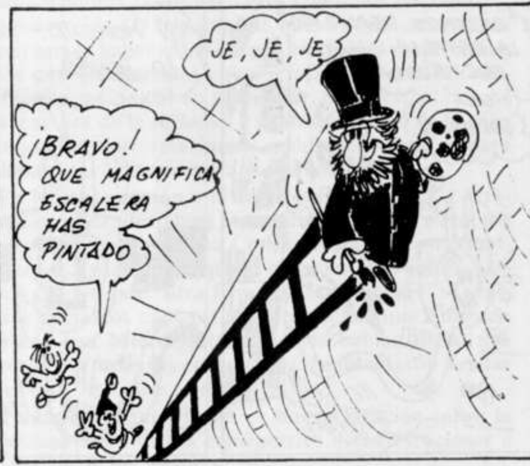
¡!
MR. CHIP.. MR. CHIP... BUG Y LOS SOLDADOS HAN DETENIDO A RAMY Y ROMO... SE LOS LLEVAN AL CASTILLO DEL EMPERADOR..
ZIUUUU
ÑIIIICCC

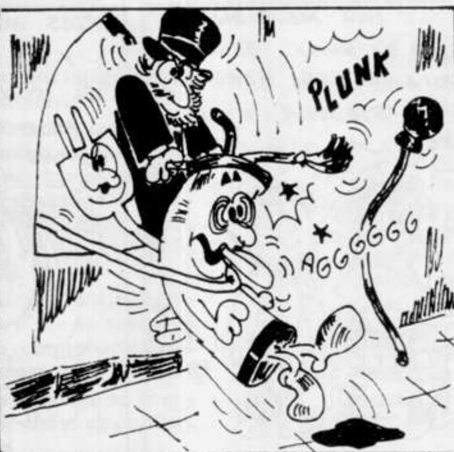
¡C'EST LE COLMO! VAMOS A IR A RESCATARLOS Y DE PASO A DAR UNA LECCION A MONSIEUR L'EMPERADOR...

MUY BIEN.. YO ME QUEDARÉ AQUI VIGILANDO LA CASA Y... ¡GLUP!...

.. AUNQUE BIEN PENSADO.. LO MEJOR ES QUE VAYAMOS TODOS JUNTOS A BUSCAR A RAMY Y ROMO

CHSSST.. DETRAS DE ESTE MURO ESTÁ EL CASTILLO.. HABRÁ QUE IR CON MUCHO CUIDADO..

HABRÁ QUE ESCALAR EL CASTILLO HASTA AQUELLA VENTANA
ESO ES COSA MIA

JE, JE, JE
¡BRAVO! QUE MAGNIFICA ESCALERA HAS PINTADO

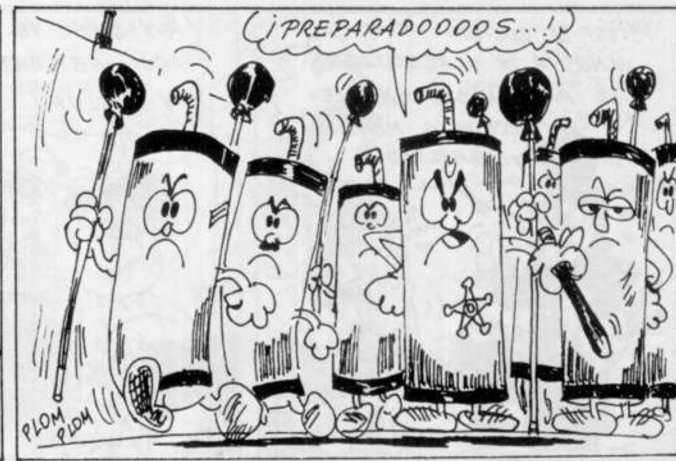
ERRAZQUIN/'84

PLUNK
AGGGGGGG

..Y AHORA VAMOS A VER QUE LE PASA AL EMPERADOR..¡ALLONS!

















Y... YA MAS TRANQUILOS Y A SALVO DEL EMPERADOR ..NUESTROS AMIGOS...







CARTA BLANCA



He de felicitaros por la nueva revista que he recibido. Pienso que ha mejorado bastante, respecto a la anterior. Leí en el número II de Commodore World que a los que les faltara los tres últimos números os escribiera. También leí que en esta nueva revista junto con la vieja, Pere Masats ha ayudado a que la revista mejorara constantemente y por eso le doy las gracias. MUCHAS GRACIAS, PERE MASATS.

Espero que la revista siga igual de bien que ahora y que vaya mejorando constantemente.

EMILIO ROCAFORT RIAZA
URBANIZACION TORRES SAN LAMBERTO
CHALET Nº 17-A
ZARAGOZA

Pasamos el truco que nos envías a Magia... y sigue enviando piropos.

El Director General de Microelectrónica y Control, D. Carlos Domenech, nos ha remitido la siguiente carta para su publicación.

Señor Director:

Nos ha producido sorpresa y preocupación la publicación en el nº 3 de la carta de D. Artemio González como "CARTA DEL MES", y casi diría que nos ha dolido más todavía la contestación y el comentario sin firma que añade la revista.

A través de nuestro distribuidor en Tenerife, conocemos al Sr. González y sabemos que es un gran entusiasta del tema "microordenadores", un asiduo lector del Club Commodore y un buen amigo. El Sr. González tiene efectivamente UNA CPU COMMODORE, PERO NO UN EQUIPO COMMODORE, ya que la Unidad de Discos que él acoplo es COMPUTHING y la impresora es CENTRONICS.

Los más antiguos de la familia COMMODORE saben perfectamente que dicha unidad de discos no ha sido nunca comercializada por MICROELECTRONICA Y CONTROL, S.A., es incompatible en Software con las unidades de disco de COMMODORE y consecuentemente con el Software de MICROELECTRONICA Y CONTROL. A los pocos que están en una situación parecido a la del Sr. González les ayudaremos con nuestra mejor voluntad, pero sólo en la medida que nos permita nuestro remoto conocimiento de unos periféricos que no son nuestros.

Permítanos insistir, Sr. Director, en la necesidad que tiene todo redactor de estar muy bien informado para evitar emitir falsos prejuicios, aunque los emita de buena fe. Como les hemos dicho, la contestación de su revista a la carta nos ha dolido y adolece la falta de información.

Atentamente,

MICROELECTRONICA Y CONTROL, S. A.

Estimado amigo:

Gracias por su carta que reproducimos íntegramente y a la que le ruego me permita contestar.

Creemos que generalmente intentamos estar bien informados (salvo posible error humano al que todos estamos sujetos). En varias ocasiones hemos recibido un par de cartas "peligrosas" que afectaban a distribuidores y que hemos comprobado con ambas partes implicadas deshaciendo el entuerto ya que en ambos casos se trataba de un malentendido.

Es cierto que D. Artemio González no nos especificaba en su carta las "peculiaridades" de su equipo, y que nosotros no realizamos ninguna comprobación previa a su publicación por varias razones.

1) El problema que este lector nos exponía no es una originalidad —ni nosotros lo consideramos un insulto a Commodore ni a Microelectrónica y Control— ya que es común a los primeros modelos de la mayoría de las marcas: la literatura está agotada o es insuficiente; el soft está agotado y a las casas de soft les es ruinoso su reedición; etc., etc.

2) Asimismo, tampoco es una situación especial "a la española" ya que es idéntica a la de cualquier otro país. En los colegios británicos en que tienen equipos "antiguos" tienen que montarse su propio tinglado (y es donde de verdad los chiquillos aprenden a desarrollar el ingenio y a "destripar" un ordenador) por la falta de material auxiliar.

3) La carta publicada fue una de tres que nos llegaron sobre el mismo tema —razón por la que se llamó "Carta del Mes", al igual que la que hoy publicamos es haciéndonos eco de bastantes quejas similares que hemos tenido— tanto por carta como por teléfono.

Lamento que le haya dolido nuestra contestación ya que creemos haber sido siempre altamente respetuosos, tanto con Commodore en general, como con Microelectrónica y Control en particular. Si no estuviéramos convencidos de la seriedad y profesionalidad de la casa Commodore no estaríamos publicando una revista dedicada exclusivamente a sus productos.

En la contestación a la carta publicada de D. Artemio González, decíamos claramente que esta situación no era peculiar de la casa Commodore, sino —textualmente— "es la misma que la de todos los equipos pioneros de cualquier otra firma de ordenadores". Igualmente decíamos en nuestra contestación, que habiendo hablado con Microelectrónica, nos habían dicho que "atenderán todas las consultas (...) e intentarán ayudar en todo lo que puedan".

Finalmente, quisiera hacer una aclaración sobre la responsabilidad de los comentarios o contestaciones a las cartas de los lectores.

El responsable de las contestaciones o aclaraciones técnicas y profesionales es nuestro director técnico, D. Diego Romero. Los comentarios de tipo general, así como las notas editoriales que aparecen generalmente en la página 4 son responsabilidad directa mía.

Ha sido nuestra intención, desde el principio, dar a la revista un espíritu de equipo y de gran familia, y como tal se está funcionando, razón por la que se ha evitado el nombre de Director, Editor o Redactor Jefe que implica automáticamente una jerarquía, y se ha optado por el de coordinador que es un papel más anónimo, pero más cordial y acorde con ese espíritu y el que honestamente creo estar desempeñando.

Espero que estas líneas desvanezcan cualquier malentendido que haya podido producirse en nuestra anterior contestación.

Un cordial saludo.

María Lopez-Morán.



...Y... SEAMOS

VIC-20 ó C-64

Lo primero, felicitaros por vuestra revista.

Tengo un Commodore VIC-20, y estoy pensando en cambiarlo por un C-64, ya que el VIC es un poco pequeño en su memoria, me gustaría que me aconsejarais sobre este cambio.

También me gustaría que me dijerais si los textos de instrucciones de los programas en cartucho, están traducidos al español, y si es así dónde conseguirlos.

Respecto a sugerencias sólo tengo una, ya que la revista roza la perfección. Yo he pensado que se podría formar una sección sobre programas educativos, a nivel de enseñanza media y superior.

En otro orden de cosas, también estoy haciendo un programa de juegos, si os hacen falta cuando lo termine os lo mando.

GUSTAVO AYALA LOPEZ
AVDA. STA. COLOMA, 35
STA. COLOMA DE GRAMANET
BARCELONA

Indudablemente el C-64 está reconocido en todo el mundo como uno de los mejores personales y con el que, con un poco de paciencia se pueden hacer virguerías. La decisión debes tomarla tú mismo, pero... si crees que el VIC se te queda corto.

Casi todos los programas en cartucho traen instrucciones en inglés. No hay reglas. Cada firma comercializadora actúa a voluntad. El tema de programas educativos es un tema que nos interesa enormemente y al que queremos dedicarnos bastante a fondo. De momento no es tarea fácil por lo poco desarrollado que está el tema aquí en España. Si lees el editorial, verás que en el próximo número anunciaremos las bases de un importante concurso escolar cuya finalidad principal es, precisamente el desarrollo de la informática educativa.

Esperamos tu programa

CARTA DEL MES

Soy suscriptor de la revista "Commodore World" desde su primer número. Mi sorpresa ha sido el ver los listados, o mejor dicho el no verlos, ya que la impresión es tan pequeña que ha sido necesaria una lupa para poder seguirlos, además de algo de conocimientos e imaginación para suponer algún que otro signo, punto y coma, dos puntos...

Una de las cosas que más me gustó en el primer número y me decidió a la suscripcón, fue la claridad de los listados. Ya no puedo retirar mi suscricpión, pero si los listados siguen así, es seguro que no la renovaré.

Me parece muy bien que quieran vender cintas con los programas que se publican en la revista y aunque el precio me parece un poco caro, quizás alguna vez pida alguna, pero si pretenden obligar a la gente a comprar las cintas haciendo los listados ilegibles, creo que van por mal camino.

Pasando a otro tema, quisiera hacer un par de sugerencias, la primera consiste en que sería interesante la publicación de un índice detallado de todos los temas aparecidos en los 15 números de "Commodore Club", así como un índice anual de "Commodore World", ya que no siempre es fácil encontrar un artículo determinado. También considero muy útil la publicación de una sección de "Fe de erratas" mejor que esos pequeños recuadros del "Bus" que a veces se traspasan entre las hojas.

La segunda sugerencia se refiere al juego "Tiro al Blanco" en su versión Vic 20 y publicado en el número uno. El juego funciona bien, sólo que es muy aburrido debido a su lentitud que lo hace desesperante, por lo que le he añadido una línea de programa que lo convierte en un juego mucho más interesante, se trata de la siguiente:

```
395 POKE 37879, 5
```

Espero que al menos consideren estas sugerencias, sobre todo la de los listados, ya que creo que marcará la diferencia entre una gran revista y unas hojas sin apenas valor.

Les saluda atentamente,

JUAN CARLOS GARCIA ALCARAZ
C/OBISPO ORBERA, 37-7-1
ALMERIA

En nuestras notas editoriales del pasado número ya aclarábamos que para una máxima garantía de corrección en los listados de los programas, estos no los pasamos por composición de imprenta sino que los reproducimos directamente de la copia de la impresora.

En nuestro número —2— de marzo, debido, quizás, a un malentendido con maquetismo e imprenta, redujeron indebidamente la reproducción. Ya se tomaron medidas para que esto no vuelva a suceder y, como ya habréis visto en el número —3— de abril, este problema está subsanado, pues ahora se lista un formato en impresora para reproducir directamente a tamaño de columna y media.

Con la venta de cintas no se pretende hacer un negocio sino dar un servicio extra al lector que lo desee. Ha sido una pura y desafortunada coincidencia que este servicio comenzara a anunciarse en ese nº 2.

Gracias por la sugerencia de los índices. No se nos había ocurrido pero es una gran idea. Pediremos permiso a MEC para confeccionar un índice de la colección de "Club Commodore". En lo que se refiere a Commodore World, teníamos previsto la preparación de unas tapas anuales para coleccionar la revista, por lo que recogemos tu sugerencia de añadir un índice anual a las mismas.

La sección "Meteduras de pata", "comenzada" en el número 3 creo que responde a tu sugerencia de "Fe de Erratas".

Finalmente, gracias por la mejora de "Tiro al Blanco".

Los siguientes commodorianos desean todo tipo de información sobre libros especializados, programas, cintas, video-juegos, precios, etc., para el VIC-20.

GABRIEL MONTOLIU ROCA
CALDERS 140 PISO
SABADELL

LLUIS MIGUEL VERDU
C/MALLORCA, 132, 3-1
BARCELONA-36

VICENTE MARTIN JUANES
C/ANTONIO DE LA CODENA, 4, 4º
PLASENCIA

XAVIER PRAT I BURDO
C/RABAL STA. EULALIA S/N.
STA. MARIA D'OLO (BAGES)
BARCELONA

Comerciantes y distribuidores molestaros y enviar información a ellos y a nosotros.

EN CINTA... EN CINTA... POR FAVOR

Ante todo quiero mandaros un saludo, y manifestaros el gran acierto que habéis tenido con nuestra revista "Commodore World" y así como un recuerdo a nuestra entrañable "Club Commodore", sin la cual no hubiera sido posible su existencia. Y ahora, sin más preámbulos, paso a comentaros varias cosas.

Quisiera pediros que en próximas ediciones dediquéis algún artículo a los Sprites: Cómo crearlos, cómo hacer que varios distintos aparezcan en pantalla, cómo hacer que se detengan en un lugar determinado de la misma para poder combinarlos creando un dibujo mayor, etc., ya que el Manual del Usuario del C-64 (que es el que poseo) es bastante deficitario en este aspecto, no estando bien explicando en muchos puntos, lo cual ha hecho que tenga muchos problemas para su creación y encadenación a un programa.

Y ya, por último, quiero indicaros que me gustaría mucho poder mandaros colaboraciones, pero tropiezo con el inconveniente que no dispongo de impresora, por lo que sólo podría mandaros el programa en cinta, no en listado. Espero que en vuestra contestación me indiquéis si los puedo enviar o no de esa forma.

Bien, ya sólo me queda despedirme hasta otra. Un abrazo.

ANTONIO SANCHEZ AGULLO
C/POLIT, 14
ELCHE
ALICANTE

Ya hemos dedicado algunos artículos a los Sprites y en este número sale otro "Anímate con los Sprites"

PREGUNTONES

Preferimos que nos envíes las colaboraciones en cinta: esto nos facilita el trabajo de comprobación y listado, o sea, que tu problema no es problema sino todo lo contrario.

¿SPRITES EN EL VIC?

Una petición: Me parece que sería interesante algo referente a otros lenguajes para VIC, tengo entendido que existe un FORTH (¿cómo conseguirlo?)

Una pregunta: ¿Es posible definir de algún modo un Sprite en el VIC? Véase el cartucho JUPITER LANDER.

Otra petición: ¿Podéis hablar de las incompatibilidades de los datasettes en el intercambio de cintas? o bien ¿es top secret?

Otra pregunta: ¿Existen ya juegos de aventuras tipo Scott Adans en España?

FRANCESC CASTANYER
(DIRECCION EN MAGIA)

Tenemos pensado dedicar algún espacio al Forth, pero como no es un lenguaje de mayorías, sólo aparecerá en algunos números. Se han publicado unos artículos sobre Forth en los números 13 y 14 de Club Commodore que, como decimos, pensamos continuar. Lo del Jupiter Lander, es algo así como un "SPRITE" formado por cuatro caracteres que los mueves a la vez, pero el Vic no tiene Sprites como tal.

¿Incompatibilidades en cintas?... Será cosa del ajuste de la cabeza, pues todos los Vics graban igual. Si te refieres a intercambios entre VIC y C-64.., lee el próximo número.

Los juegos tengo entendido que hay una casa que los va a traer (son rumores de momento... pero de bastante "buena tinta").

ELEMENTAL, QUERIDO WATSON

Desearía saber cómo se realizan las siguientes sentencias:

PRINT "[SHFT CLR]"
PRINT "[CTRL 8] [SHFT Q]"
PRINT "[CRSR DN]"

es decir, cómo se realizan en el listado de un programa, por ejemplo, en el primer caso se escribe Print y luego se aprietan las teclas SHIFT y CLR/HOME.

JORGE VIDAL ANALL
C/ALTA SAN PEDRO BLOQUE B, Nº 11
TORREDEMBARRA
TARRAGONA

Esto está explicado en la "Clave para interpretar los listados", que aparece en todos los números, en este concretamente en la pág. 7.

AIRE ACONDICIONADO PARA C-64 Y COLIRIO PARA EL USUARIO

Ante el conocimiento de la existencia de micros, que al estar enchufados largo tiempo acaban quemados.

Desearía saber el tiempo que puede estar encendido en mi caso particular un C-64

También, si con un ventilador enfocado al transformador se puede aumentar dicho tiempo y cuanto.

J. F. FERNANDEZ AREVALO
AVDA. MEDITERRANEO, 39, 3, 1
MADRID-7

¡Hasta que se te salten los ojos, guapo! No sabemos de ningún C-64, que se haya quemado. El que utilizamos en la revista trabaja, a veces, durante más de 40 horas sin descansar ni fallar. Si prefieres ponerle un ventilador hazlo...

Repecto a lo que nos preguntas en tu otra carta sobre el Cartucho Pascal, la hemos enviado a Microelectrónica y Control para que te contestaran personalmente lo más rápido posible, por lo que esperamos que ya lo tengas solucionado.

¡Y DICE QUE NO PREGUNTA!

Mi nombre es José Manuel: Como es la primera vez que os escribo a pesar de ser socio del club hace un año, tengo algunas cosas que preguntar: Y como decís que hay que preguntar...

1º Para ordenador VIC-20, como se realizan las raíces enésimas (cúbicas, 4ª, 5ª, etc.)

2º Teniendo el expander model VIC-1020:

2.1. Se pueden poner dos cartuchos de memoria iguales (dos de 8 k o dos de 3k, etc) ¿Qué sucedería si se pusieran?

2.2. ¿Cuál es la mayor cantidad de memoria (en cartuchos) que se le puede poner al expander? ¿Y cómo estaría distribuida en cartuchos?

2.3. ¿Los tres k de memoria que vienen con el superexpander sirven para programación Basic?

2.4. ¿Con el superexpander se puede poner un cartucho de 3k (en el expander) y se amplía la memoria?

3º Las cintas de colaboraciones que mandemos ¿nos serán devueltas con otro programa? ¿Este programa lo podemos elegir nosotros? Y si es así ¿pueden ser juegos buenos como comecocos, frogger, etc?

Me despido de vosotros deseándoos buena suerte en la revista.

JOSE MANUEL RIVAS TORRES
TORRECEDEIRA III, 8º dcha.
VIGO-2

1. R= $\sqrt[n]{X}$ => R= X ↑ (1/N). Esto sirve para todos los Commodores.

2.1. Se "grillaría" tu VIC.

2.2. 1 de 16k, 1 de 8k y 1 de 3k, simultáneamente, por lo que tendrás tanto, 32k RAM con la del ordenador.

2.3. Sí.

2.4. Otro más ¡NO, POR FAVOR!

3º Serán devueltas con la mayor rapidez posible.

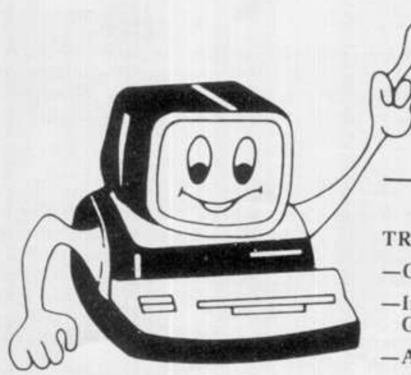
Para elegir tendríais que pasar por aquí y escoger uno entre nuestra colección. Normalmente lo elegimos nosotros.

No podemos enviaros el "comecocos" porque no es de nuestra propiedad y nunca nosotros, Commodore World, robamos Copy rigths

FINAL

Y, oye, tu no habrás hecho preguntas nunca, pero hoy te has desquitado... ¿Alguna más, hijo?

¿QUIERES SACAR EL MAXIMO PROVECHO A TU COMMODORE 64?



TRONIK, tu amigo informático te ofrece:
—Curso a distancia del COMMODORE.
—Iniciación a la informática y BASIC del COMMODORE en nuestras aulas.
—Accesorios para el COMMODORE y VIC 20.
—Libros y revistas.
—Alquiler de cartuchos de juegos debajo de Accesorios.

Los 25 primeros tickets recibidos tendrán el obsequio de un programa para su COMMODORE 64

Para mayor información, dirígete a TRONIK
Bigay, 11-13 - Telf. 212 85 96 - Barcelona-22

Nombre ....
Dirección ....
Población .... Telf. ....
Provincia ....



MARKETCLUB

Servicio gratuito para nuestros lectores particulares. Empresas 300 ptas., por línea.

MERCADILLO

(Ref. M-9)

• Desearía comprar o intercambiar programas de juegos o con diversas utilidades para el VIC-20 no ampliado (cinta). Carlos Plaza Hernández. Calle Mestre Nicolau, 37-41. Entlo. 3.ª. Teléfono: (93) 380 13 17. **BADALONA** (Barcelona). Preferible llamar de 13 a 16 ó de 22 a 23.

(Ref. M-10)

• Ruego toda información sobre programas y aplicaciones de 3032 así como monitor para lenguaje máquina con literatura en francés o español. Artemio González Pérez. Calle Fernando Arocena Quintero, 1, Barrio de Buenos Aires. **SANTA CRUZ DE TENERIFE**. Teléfono: (922) 22 42 44.

(Ref. M-11)

• Vendo microordenador VIC-20 sin cassette, por 34.000 ptas. y regalo más de 70 programas de todo tipo, entre ellos, control stock, ajedrez (2), cuentas comerciales, archivo QSL con presentación en pantalla de la misma y posibilidad de imprimirla, juegos, etc., etc. Interface CW-RTTY autoconstruido, funciona a la perfección y vale para cualquier ordenador, se incluyen programas de CW y RTTY por 10.000 ptas. Manuel González Lombardía EA1CMC. Calle Gabriel y Galán, 4-2º A. **PEÑARANDA DE BARCAMONTE** (Salamanca).

(Ref. M-12)

• Intercambio programas de todo tipo para el VIC-20. Regalo programas a todo aquel que pida mi lista. También vendo ampliación de memoria 16K por sólo 12.000 pesetas. Dirigirse a: Tomás Toribio. Calle Jerusalén, 5. **ZARAGOZA-9.** Teléfono: (976) 35 69 94.

(Ref. M-13)

• Si os sobra una impresora para el VIC-20 y no sabéis a quién regalársela yo estaría muy complacido de recibirla. Jorge Rodríguez Añeyro. Polígono San Pablo, Barrio C, Torre 2º 4C. **SEVILLA-7.**

(Ref. M-14)

• Desearía comprar cartucho Super Expander para VIC-20. Precio a convenir. Llamar de 2 a 2,30 ó de 7 a 11. Tel. (954) 61 41 24. Calle Gustavo Gallardo, 7-4º D. **SEVILLA-13**. Cuanto antes llamen mejor.

(Ref. M-15)

• Compro placa Indescomp de ampliación de memoria a 32 K, cartucho "Superexpander" y "monitor lenguaje máquina", para VIC-20. Luis Astrolfi Antona, Avda. 5º Nº 22. 2D. Ciudad Pegaso. **MADRID**-22.

(Ref. M-16)

• Agradecería a quien pudiera enviarme información sobre monitores a color compatibles con el VIC-20, sobre todo con precio. También si existen programas de simulación y aventuras para el VIC-20, y si así fuera, su precio, memoria que ocupan y en qué consisten. Por favor sólo por correo, no visitas a domicilio ni llamadas telefónicas. Mario Díaz Gañán. Comandante Zorita, 25, 4ºC **MADRID-20**

(Ref. M-17)

• Intercambiaría programas para VIC-20, con 3K alta resolución y ponerme en contacto con universitarios o estudiantes de informática. José María Martínez García. Apartado 303. **CARTAGENA.**

(Ref. M-18)

• Vendo algunos interfaces para VIC-20 y 64, para RTT/CW. Totalmente eficaces tanto en facilidad de sintonía como en respuesta. Filtro de 425 Hz para comerciales. Precio 20.000 Ptas. Regalo programas RTTY/CW y archivo QSL. Solicitad información. EA 1BDD. Arturo Sánchez Galán. c/Generalísimo, 9. Teléfono 42 ó 43 de Solosancho. **(AVILA).**

(Ref. M-19)

• Compraría ampliación de memoria de 16 ó 32 K. Juan Mª. Tel. 972/570 497. C/Santa María, 26. Banyoles **(GERONA).**

(Ref. M-20)

• Intercambio programas para VIC-20 de todo tipo: Juegos y utilidades. También específicos para radioaficionados: Orbitas, QTH, locator, etc.

Alfonso Moraleda Pérez (EA4CAI) Clara del Rey, 81 **MADRID-2**. Teléf.: (91) 415 02 24

(Ref. M-21)

• Se compra cartuchos de expansión para el VIC-20 y cintas de juegos para el mismo, llamar o escribir a Joan Serra Conde, C/Blanch, nº 3, **MALGRAT** (Barcelona), Teléf.: 761 10 17.

(Ref. M-22)

• Aceptaría ofertas. Si pudieran darme informes de programas para el C-64. Albert Godo Falco. Apto. de Correos nº 143 **IGUALADA** (Barcelona). Teléf.: 803 01 25. Edad 18.

(Ref. M-23)

• Usuario de C-64, estoy interesado en el intercambio de experiencias y programas de todo tipo. José Alberto Gaforio. Plaza Reyes Magos, 1. Teléf.: 251 75 18. **MADRID.**

(Ref. M-24)

• Dispongo de programa para VIC-20 +"8/16 Kb" ó "8/16+3Kb" que añade 35 (treinta y cinco) NUEVOS comandos al Basic del VIC, dos tipos de letras: 40 columnas por 24 filas y 20 col. por 24 fil., también alta resolución de 160 por 192 pixel. Todo ello de uso simultáneo sin cambiar de modo. Acompaño con el programa las explicaciones y programa demostración. 3.000 ptas.
Guillermo Falgueras Cano. Apart. 211- **LA LINEA** (Cádiz). Teléf.: 766 999. Preguntar por Guillermo y 766 532 preguntar por Juan.

(Ref. M-25)

• Tengo un Commodore 64, y soy agente libre de seguros, desearía saber si se han desarrollado programas específicos para esta actividad o algunas aplicaciones.
Luis Angel Gutiérrez Díaz - Angela Peña Calvo. C/Diego de Velázquez, 6-4º Dcha. Teléf.: (941) 23 75 22. **LOGROÑO.**

(Ref. M-26)

• VENDO:

• TRS 80 compuesto de teclado con 16 K, monitor B/N, impresora rápida térmica, manual en castellano y cassetes de programa, 50.000 ptas. (1979).

• SINCLAIR ZX81 compuesto por teclado profesional 32K, impresora ZX, unidad de sonido, manuales en castellano, 30.000 ptas. (marzo 1980).

• VIC 20 compuesto de teclado con 16K, manuales en castellano, cartucho juegos, 1ª parte. Introducción a la informática, 30.000 ptas. (agosto 1983).

• CASIO PB100 compuesto de unidad central, 1510 pasos memoria, interface cassete, impresora térmica, manuales en castellano, 20.000 ptas. (agosto 1983).

Llamar al 311 27 06, preguntar por Carlos Infante, de 12 a 16 horas.



CLUBS

(Ref. C-1)

• Club de amigos del VIC-20 y Commodore-64. Ramón P. Serna Soler. Foto Estudio 2. Plaza de Sombrereros, 2. **PALMA DE MALLORCA**. Tel. 21 31 62.

(Ref. C-2)

• Desea información sobre clubs en existencia y gente que quiera formar uno. Agustín Quevedo Velasco. Oña, 55-4° B. **MADRID-34**. Tel.: 202 94 28.

(Ref. C-3)

• Desea información sobre clubs en existencia y gente que quiera formar uno. Pablo Nistal Alonso. Empecinado, 46. 6° B. **MOSTOLES**. Tel. 645 21 70.

(Ref. C-4).

• Club de programación Alaiz, en Pamplona. Está dirigido a estudiantes de B.U.P. y C.O.U. de esta ciudad, y el fin que tiene es la introducción en el mundo de la informática y programación BASIC. No tiene, por lo demás, ningún fin lucrativo. El club propiamente posee equipos COMMODORE 64, VIC-20, y diversos periféricos (impresora, unidad de disco y cassette), CLUB ALAIZ. Pza. Monasterio Santa Gemma, s/n. Tels.: 254480-257704. **PAMPLONA.**

(Ref. C-5)

• Club de Commodore de **Albacete**. Fernando Martínez Guerrero. La Roda, 39 5° D. **ALBACETE.**

(Ref. C-6)

• Círculo Durensan Vigues. **VIGO**. Venezuela, 48. Entrechan. Tels.: (986) 410683-422519.

(Ref. C-7)

• Un grupo de usuarios de la Comunidad Autónoma de Murcia intenta crear un Club de Usuarios de VIC-20 y C-64 en esta región, sin fines lucrativos. Interesados llamar a los teléfonos (968) 80 23 64 (Mañanas, preguntar por Paco); (968) 26 72 62 (Tardes, preguntar por Fernando). Francisco Sáez Soto. C/Virgen del Rosario, 28. **ALCANTARILLA. MURCIA.**

(Ref. C-8)

• Deseo contactar con interesados en formar un club de microordenadores en Barcelona e intercambiar experiencia Domenec Garrofe. C/Aragón, 575, 4° 2° **BARCELONA-26** Telf. 348 00 39.

(Ref. C-9)

• Interesados en la creación de Club de Usuarios en Girona escribir a "Apartado de Correos nº 580" de **GIRONA.**

(Ref. C-10)

• Xabier Prat i Burdo que vive en c/ Rabal Sta. Eulalia s/n. de Sta. María Dólo (**BAGES**) Barcelona quisiera saber si existe algún Club Commodore cerca de Manresa y solicita información sobre todo lo relacionado con Modems.

(Ref. C-11)

• Os rogaría me comunicarais si existe algún Club Commodore en mi ciudad (**GUADALAJARA**) o en **ALCALA DE HENARES** (donde trabajo). También agradecería información acerca de accesorios, publicaciones, etc., así como de distribuidores de Madrid que trabajen con el C-64. Miguel Angel Capeña Alonso. c/ Tabernero, 16-3º I. Telf. (91) 22 67 57.

(Ref. C-12)

• Desearía información sobre la existencia de algún Club Commodore en Somaén o sus cercanías. Carlos Cabrera. Camino de las Eras, s/n. Somaén (Soria).

BOLSA DE TRABAJO

OFERTAS

(Ref. BO-1)

• **PROGRAMADORES C-64**. Si conoces a fondo el Commodore 64 y quieres trabajar, envíanos tu curriculum vitae y una foto reciente al apartado de Correos de Madrid 10.048. Contestaremos todas las solicitudes.

DEMANDA

(Ref. BD-1)

• Realizo procesamiento estadístico de datos. Tfno: 91-273 20 76. Pedro Caminal.

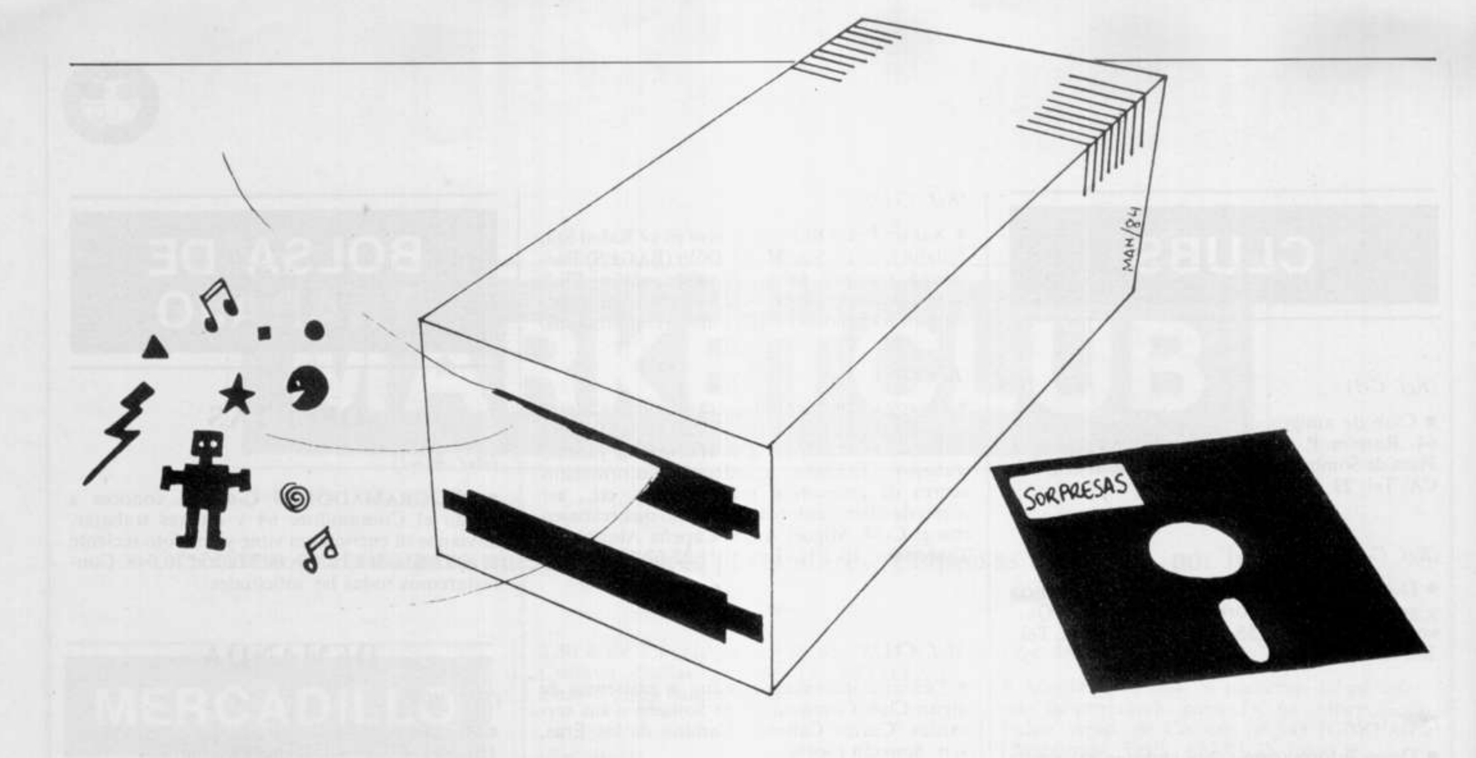


—Commodore World, dígame.

—¡¡**Se me ha perdido el boletín de suscripción!!**

—No se preocupe, por teléfono también puede suscribirse llamando al número (91) 259 54 78.





La Caja de los Misterios

¿Te encuentras un poco confuso a cerca de esa caja de misterios llamada la unidad simple de discos 1541? Este artículo aclarará un poco este tema y te ayudará a aprovechar al máximo tu unidad de discos

La mayoría de los principiantes descubren rápidamente las desventajas de utilizar cintas para cargar y salvar programas, con el resultado de que pronto deciden comprarse una unidad de discos. La unidad simple de discos 1541 resulta ser un complemento potente y barato para cualquier VIC-20 o C-64, pero esta ventaja trae sus complicaciones.

La lectura del manual no ayuda mucho ya que resulta tan complejo como la misma unidad, y por lo tanto intimida un poco a los principiantes. La finalidad de este artículo es la de presentar una explicación sencilla y detallada de las características y operaciones de uso más frecuente de la 1541, orientadas específicamente al que use la unidad de discos por primera vez.

Abre Tu Manual...

Usando este artículo, la unidad de discos, el manual de usuario de la unidad 1541 y el disco de demostración, serás capaz de aprender las técnicas de trabajo con disco en programas BASIC. Aprenderás a desembalar, conectar y usar tu unidad de discos, y a leer los directorios de los discos. Te enseñaré a cargar programas de discos pregrabados y cómo salvar programas en discos nuevos.

Finalmente, aprenderás a utilizar el programa soporte del DOS ("wedge" o cuna), y cómo usar las utilidades incorporadas en la 1541 para comprobar, borrar y cambiar el nombre de los programas en disco. Cuando termines de leer este artículo, sabrás todo lo que hay que saber sobre las operaciones más importantes en disco, y podrás entender el contenido del manual lo suficiente como para empezar a aprender las demás operaciones.

Tarde o temprano, dentro del mundo de la electrónica, te darás cuenta de la verdad del proverbio "Cuando todo lo demás te falle, consulta el manual". Sin embargo, resulta bastante más difícil leer un manual que leer el "Hola", pero incluso el manual más complicado contiene casi todo lo que necesites saber sobre el funcionamiento del equipo.

Aunque queda lejos de ser perfecto, el "Manual del Usuario de la Unidad Simple de Discos 1541 del VIC-20", es mejor que muchos manuales de equipo, y es la mejor fuente de información recopilada sobre la 1541. Asumo que dispones de un manual al que referirte cuando leas este artículo, así que intentaré evitar las repeticiones de la información que contiene el manual.

El manual que yo tengo se identifica en la página del título como P/N 15400018-02, y en la página que nombra la editorial con fecha de septiembre de 1981. Si el tuyo es diferente, no te preocupes; podrás seguir estas aclaraciones.

Toma tu manual y mira cuidadosamente el índice en la página 1. Repasaremos los contenidos de los capítulos 1-4, de una forma más comprensible e incluyendo unas explicaciones importantes. La información más complicada de los capítulos 5-9 resulta demasiado avanzada para los fines de este artículo.

Velocidad y Capacidad

El capítulo 1 te dará una base excelente para comprender este artículo y el resto del manual. Los discos resultan realmente rápidos. Lo comprobé con un programa de 64K que cargué primero de cinta y luego de la 1541, los resultados fueron: cinta, 311 segundos; disco, 45 segundos. ¡No está mal! Pero pienso que hay unidades de discos más rápidas que la 1541; mi PEDISK II cargó el mismo programa en cuatro segundos. De todos modos la 1541 es una gran mejora sobre el DATASSETTE.

Sólo hay tres características del disco que interesan al principiante (páginas 5 y 7). La 1541 utiliza discos normales sectorizados por programas, de una sola cara y simple densidad. Esta es una buena noticia, ya que normalmente son los más baratos. Puedes usar otros tipos de discos, pero pagarás dinero por una calidad que la 1541 no puede aprovechar.

La capacidad de un disco tiene unos

VIC-20
C-64

algunos de nuestros programas para los ordenadores personales

commodore













B.M.

BASIC MICRO-ORDENADORES, S. A.

AVD. CESAR AUGUSTO, 72

Tlfs. 23 56 82 y 22 65 44

ZARAGOZA-3

límites bien definidos. Puedes guardar 144 programas en un disco, siempre que ocupen menos de 174.848 bytes. Al quedar salvados ese número de bytes, el disco se considera lleno, aunque los programas grabados sólo sean unos pocos. Si intentas salvar un programa que sobrepase este límite, la 1541 enviará un mensaje de error.

Conexión

En este capítulo existen varias omisiones. Conviene que el ordenador se conecte en último lugar para que la unidad de discos y la impresora ejecuten correctamente la secuencia de encendido.

En el programa soporte del DOS (cuña), no menciona otros dos comandos del C-64:

↑NOMBRE = LOAD "NOMBRE",8 + + RUN
%NOMBRE = LOAD "NOMBRE",8,1
←NOMBRE = SAVE "NOMBRE",8

Cargando el Programa

El capítulo 3 del manual contiene mucha información, pero puede resultar un poco difícil sacarla toda. Para que esta operación resulte un poco más fácil, hablaremos de varios temas en un orden diferente y explicaremos unas cosas que no son tratadas en el manual. Primero, vamos a mirar los programas en un disco pregrabado —el Disco de Demostración VIC 1541 Test/Demo que acompaña la unidad de discos 1541.

Si has seguido las instrucciones del capítulo 2, el ordenador y la unidad de discos tienen que estar encendidos, esperando el siguiente comando. Siguiendo las instrucciones de la página 11 para la inserción del disco, introduce el disco Test/Demo en la unidad de discos y cierra la puerta.

Tal y como sale de la fábrica, el disco Test/Demo viene con varios programas grabados junto con un directorio. De hecho, cada disco incluye un directorio como parte de la operación de inicialización en una unidad Commodore. El directorio indica lo que se haya grabado en disco, si es que hay algo grabado, y su lectura es una operación importante y frecuente.

El directorio puede ser cargado y listado, igual que un programa en Basic. Para cargar un directorio de disco, se teclea LOAD, seguido por el signo del dólar entre comillas, una coma y el número 8. A continuación, se teclea la tecla "Return". Por lo tanto, con el disco Test/Demo cargado en la unidad, se teclea:

LOAD "$",8

Se vuelve a pulsar la tecla "Return", y debe de aparecer un mensaje de cargar, seguido a los pocos segundos por un mensaje "READY"

Si aparece otro mensaje, será señal de que han introducido incorrectamente el disco. El LED rojo de la 1541 se enciende y se apaga para indicar que algo va mal. Si esto ocurre, comprueba el disco y vuelve a probar con el comando Load.

Una vez cargado correctamente, $ se lista, y se ve en seguida que no se parece en nada a un programa normal en Basic; debe parecerse al listado en la Figura 1. A continuación, se presenta una explicación del significado del directorio.

El cero que aparece al principio de la primera línea indica que el directorio procede de la unidad de discos 0. Si el usuario dispone de una unidad doble como la 4040 o la 8050, una de las unidades se llamaría 0 y la otra 1; este primer carácter distinguiría el directorio de cada una.

El nombre de campo invertido que aparece entre comillas en la primera línea es el nombre del disco, que le fue asignado cuando el disco se inicializó para una unidad de discos Commodore. Igual que los nombres de los programas, éste puede contener hasta 16 caracteres.

El ZX que aparece en la primera línea constituye el identificador de dos caracteres del disco, que fue establecido cuando el disco fue formateado. El nombre del disco y el identificador constituyen dos maneras totalmente independientes de identificar el disco. A diferencia del nombre, el identificador se graba muchas veces en el disco y está estrechamente relacionado con las operaciones de la 1541. En la medida en que sea posible, no se debe de asignar el mismo identificador a más de un disco.

El 2A que finaliza la primera línea indica la versión del DOS bajo la cual se formateó el disco. Si no sabes lo que esto significa, no te preocupes, de momento, no te hace falta.

Las siguientes líneas del directorio contienen información sobre los 15 programas grabados en el disco. El número al principio de cada línea indica el número de "bloques" que un programa determinado ocupa en el disco, y puede servir para indicar más o menos la longitud del programa. Como podrás ver, "How to Use" es un programa que resulta mucho más largo que el "C-64 Wedge".

Los nombres de los programas, con un máximo de 16 caracteres, aparecen entre comillas detrás de los indicadores de bloques. Las letras PRG que siguen el nombre del programa indican que la información grabada es un programa, y no un fichero de datos. Se utilizan letras distintas como SEQ o REL, para indicar ficheros de datos de diferentes tipos. A decir verdad, lo que nosotros hemos llamado nombres de programas deben llamarse nombres de ficheros, dado que para la 1541, un programa solamente constituye otro tipo de fichero.

A medida que vayas usando tus directorios en el futuro, podrás ver que los programas nuevos añadidos a un disco no aparecen siempre al final del directorio, sino que a veces se insertan en medio. Esto se debe a que la 1541 introduce los programas donde encuentre espacio, es decir, en cualquier sitio que aparece en el disco. El directorio los lista en el orden en que los encuentra, lo cual puede resultar un poco confuso a veces.

La última línea del directorio indica cuántos bloques de almacenamiento quedan en el disco. A medida que se añaden programas, este número se va reduciendo, y a medida que se borran, se aumenta.

Para encontrar los nombres de los programas grabados en disco, se carga y se lista el directorio, igual que hemos hecho en el disco Test/Demo. El siguiente paso normalmente es el de cargar uno de estos programas introduciendo:

LOAD"xxxx",8

cuando xxxx es el nombre del programa deseado, tecleado en mayúsculas.

Existen unos detalles y unos atajos bastante útiles para cargar programas de disco, los cuales quedan bastante bien explicados en el manual. Primero lee las páginas 27 y 28 y 74 y 75.

Ahora que hemos leído como se cargan los programas de disco, vamos a hacerlo. El disco Test/Demo contiene más de una docena de utilitarios en disco que te pueden ser útiles en el futuro. Los dos primeros programas en el disco contienen unas instrucciones breves sobre el uso de los demás.

Para empezar, carga el programa llamado "How to Use" introduciendo el comando LOAD"H*",8. (Si lees el manual, sabrás que se utiliza el asterisco como pauta de acuerdo para ignorar el resto del nombre.)

Ahora ejecuta el programa para leer las descripciones de los demás, pero no cargues ninguno en este momento. Cuando pulsas la barra de espaciado, asegúrate de que la sueltas en seguida, dado que la acción repetitiva borrará alguna información importante.

Cuando llegas al final del programa, el comando LOAD"HOW PART TWO",8 se encontrará en la misma línea que el cursor. Pulsa la tecla Return para ejecutarlo, y observa el LED rojo de la unidad de disco mientras escuchas el motor; el LED se encenderá mientras se carga el programa y se apagará, pero si destella intermitentemente significa que hubo un error. Ahora ejecuta el programa que acabas de cargar y lee la información sobre los últimos programas en el disco.

Con esto ponemos fin a las explicaciones sobre cómo cargar un programa. Para que te resulte más cómodo, las instrucciones para los programas "How to Use" se presentan en la Tabla 1, y los listados para algunos de ellos aparecen en el Apéndice A del manual. Ahora estamos preparados para grabar en disco.

Grabando en disco

Un disco nuevo está totalmente en blanco. Antes de poder utilizarlo, debes darle un nombre y un identificador de dos caracteres, y hay que reservar un poco de espacio para almacenar el directorio. Esta operación se llama comunmente formatear el disco, y esto se realiza mediante una serie de comandos en el teclado.

Estos comandos (Open, Print# y New) están descritos en el capítulo 3 del manual (páginas 17 y 18). Estas descripciones aclaran que un disco en blanco puede ser formateado al introducirle a éste en la 1541 y teclear OPEN15,8,15,"NEW0:VICTORIA,BS".

El disco formateado así se llamará VICTORIA, y su identificador será BS. Si

tienes un disco en blanco, intenta a formatearlo ahora. A continuación, se carga y se lista su directorio para poder ver los resultados.

Dado que la operación de formateado significa que hay que escribir en el disco, la muesca de protección no debe estar tapada. (La muesca de protección es una franja recortada de 1/4 de pulgada cuadrada en el borde del disco que, cuando se tapa con una cinta, prohibe la escritura en el disco). Si intentas escribir en un disco protegido, el LED rojo empezará a destellar, indicando una situación de error.

Resulta increíblemente fácil salvar un programa en Basic en un disco formateado. Lo único que hay que hacer es introducir lo siguiente:

SAVE"xxxx",8

cuando xxxx representa el nombre del programa deseado. Al igual que con el comando LOAD, hay que comprender unos cuantos detalles que se explican en la sección titulada Save en la página 26 del manual.

El comando Verify, que resulta tan útil en el Datassette, también queda disponible en disco. Pero dado que es mucho más fiable salvar en disco que en cassette, el comando Verify se utiliza mucho menos con los discos. El comando Verify se explica en la página 27 del manual.

Otros Comandos del Disco

Se utiliza el comando OPEN15,8,15 cuando la operación de formatear un disco resulta un poco más complicado que lo normal. El canal 15 es el llamado canal de comandos o el canal de errores, y se puede utilizar para enviar varios comandos distintos a la 1541. El canal se abre introduciendo OPEN n,8,15, cuando n, el número del fichero lógico es cualquier número del 1 al 127. La mayoría de los usuarios hacen que n=15 para evitar cualquier confusión.

El canal de comandos tiene varias maneras para cerrarse, algunas de las cuales son bastante sutiles, y si el canal se cierra sin querer, podrías pensar que el ordenador se ha vuelto loco. Las sentencias en Basic Close, CLR, New y Run, y el añadir o el borrar líneas de un programa en Basic cerrarán el canal de comandos. Sin duda, existen otros métodos para hacerlo. Hay que tener en cuenta que un canal abierto se puede cerrar en cualquier momento, y te tienes que preparar por si esto ocurre.

Cuando un canal de comandos se encuentra abierto, los comandos se envían a la 1541 tecleando PRINT#15,xxxx cuando xxxx representa el comando en cuestión. (Si el canal se ha abierto con un número del fichero lógico que no sea el 15, ese número, y no el 15, tiene que seguir el comando PRINT#.

La mayoría de los comandos consisten en una palabra o la abreviatura de ésta, algunos son puntuaciones y otros son parámetros. Los comandos de más interés para el principiante son Scratch, para borrar un programa del disco, y Rename, para darle un nuevo nombre al programa.

Los comandos Initialize y Validate también ofrecen algún interés. Estos y otros comandos se explican en las páginas 19, 21 y 23 del manual.

Errores y Mensajes de Error

Las varias condiciones de error que pueden hacer destellar el LED rojo se listan en el capítulo 8 del manual. Existen errores de lectura, de escritura, de sintaxis y muchos más. Si esto te interesa, consulta el capítulo 8.

Cuando el LED rojo destella, la información sobre el error que ha provocado la intermitencia se encuentra en el canal de comandos. De hecho, dicho canal muchas veces es llamado el canal de errores. Al escribir y ejecutar un programa corto, se puede leer la información sobre errores, la cual puede resultar útil para diagnosticar el error. Los detalles se presentan en la página 69 del manual, dentro de la sección titulada Petición de Mensajes de Error.

Cuando el LED rojo empieza a destellar, continúa así hasta que se lee el canal de errores o cualquier otro suceso da fin a la condición de error. Como es de esperar, estos sucesos pueden ser apagar y volver a encender el ordenador, cargar un programa de disco o cargar el directorio del disco en la memoria.

El Programa "Wedge"

El programa de soporte DOS, que también se llama "wedge" o cuña, es uno de los programas más útiles en el mundo de Commodore, pero también uno de los peores documentados. Su finalidad es la de eliminar muchas pulsaciones de teclas que normalmente se necesitan para trabajar con la unidad de discos, y lo hace bien. Probablemente tu disco Test/Demo contiene un "wedge" para tu ordenador. En mi disco existen dos —un "Wedge" para el VIC-20 y otro para el C-64. El tuyo puede tener unos nombres distintos, pero seguramente se encuentran allí.

Una vez cargado y ejecutado el programa "wedge", otros programas pueden ser cargados al teclear la barra de fracción y el nombre del programa, y pulsando la tecla "return". El nombre del programa



no tiene que cerrarse entre comillas, y el asterisco o la interrogación pueden ser usados normalmente para abreviarlo.

El "wedge" permite la lectura del canal de errores al teclear > o @, y pulsando a continuación la tecla "return". Si cualquiera de los dos símbolos va seguido de un signo de dólar, el directorio saldrá en pantalla sin interferir cualquier programa almacenado en memoria. Además, cualquiera de los dos símbolos puede sustituir PRINT# cuando se usa el canal de comandos, sin tener que abrir el mismo canal.

El "wedge" se explica en la sección sobre el Programa Soporte del DOS en la página 14 del manual y en la salida del programa "How to Use". Algunas versiones del "wedge" disponen de más comandos que los que se listan en el libro. Hablar de todos ellos sale de los límites de este artículo, pero si se lo consultas a un usuario de la 1541 más experimentado, a lo mejor te puede ayudar.

Los usuarios más experimentados de la unidad de discos utilizan el "wedge" con mucha frecuencia y pocas veces usan los comandos más largos. Tú también lo puedes utilizar, si quieres. Hemos esperado hasta el final del artículo para hablar del "wedge" para que pudieras entender bien las operaciones de la 1541 antes de coger este atajo, y también porque no todos los usuarios de la 1541 disponen de un programa "wedge" para su ordenador.

A propósito, el origen del término "wedge" (o cuña) es un misterio. Algunos dicen que este nombre poco común viene de la forma del símbolo "más que" que se utiliza para llamar al programa, mientras que otros dicen que viene de la manera en que los comandos nuevos se introducen mediante una cuña ("wedge") en el intérprete Basic. Pero a mí me han dicho que ninguna de las dos cosas es cierta. Como tantas otras palabras de informática, ésta es una abreviatura. ¿Porqué la palabra "wedge"? ¡Nosotros los expertos nunca damos explicaciones! (We Experts Don't Give Explanations).

Ahora que has visto todas las operaciones básicas de la 1541 y las has consultado en el manual, podrás entender bastante bien el funcionamiento de tu unidad de discos. Sin embargo, como hemos dicho antes, sólo hemos tocado la superficie de todas las capacidades de la 1541.

El resto del manual contiene la clave del resto de las cosas que puedes hacer. Después de haber trabajado un poco con la 1541, a lo mejor te apetece conocerlas para descubrir lo que puedes hacer.

Tabla 1. Instrucciones del programa "How to Use" en el disco Test/Demo.

INSTRUCCIONES EN DISCO

Existen unos comandos adicionales que permiten al usuario teclear instrucciones cortas para la unidad de discos. Se carga y se ejecuta el "VIC-20 Wedge" para un programa VIC-20, y un "C-64 Wedge" para un C-64.

Para cargar un programa, ahora podrás teclear una barra de fracción, seguida del nombre del programa. El signo / significa cargar de la unidad de discos. Por ejemplo, se teclea /INVASION para cargar el programa llamado Invasión.

Se teclea > o @ para sacar en pantalla el estado actual de los errores. Se teclea> $ o @$ para sacar en pantalla el directorio sin borrar el programa actual.

PROGRAMA COPY/A11

El programa Copy/A11, escrito por Jim Butterfield, puede ser usado para copiar ficheros de una unidad a otra si las dos unidades van conectadas al mismo ordenador. Primero es necesario cambiar el número de dispositivo de una unidad de 8 a 9, mediante el uso del programa Disk ADDR Change.

Se carga y se ejecuta el programa Disk ADDR Change, después de los mensajes; a continuación, se carga y se ejecuta el programa Copy/A11. Esto te permite copiar del periférico 8, unidad 0 (la unidad no cambiada) al periférico 9, unidad 0 (la unidad que acaba de cambiarse).

El disco en que se va a realizar la copia no debe llevar la muesca de protección tapada, y es buena idea pegar un trozo de cinta adhesiva encima de la muesca de protección del disco que se va a copiar.

PRINTER TEST

El programa Printer Test imprime una lista de caracteres en un formato que permite una fácil comprobación de las capacidades mecánicas y electrónicas de la impresora.

VIEW BAM

El programa View BAM permite que el programador examine el contenido de los sectores que componen el mapa de disponibilidad de bloques, o el RAM. El BAM es la tabla utilizada por el DOS para identificar los bloques que han sido asignados a los ficheros en el disco.

DISPLAY T&S

El programa Display T&S permite que el programador examine el contenido de un bloque del disco, especificando los números particulares de pista y sector que identifican dicho bloque.

CHECK DISK

Con el programa Check Disk podemos comprobar si un disco nuevo que ha sido formateado es utilizable. El programa escribe en cada bloque para comprobar la capacidad de éste para almacenar información. El programa también identifica y asigna cualquier bloque no utilizable en el BAM para que no sea utilizado posteriormente por el DOS.

Se pulsa la tecla "return" para cargar la segunda parte. Después de que se cargue, se teclea RUN, se pulsa la tecla "return" y se teclea LOAD"HOW PART TWO",8

PERFORMANCE TEST

El programa Performance Test permite la comprobación de las capacidades electrónicas y mecánicas de la unidad de discos cuando sea necesario. Utiliza este programa si sospechas que la unidad ha sufrido algún daño.

SEQUENTIAL FILE Y RANDOM FILE

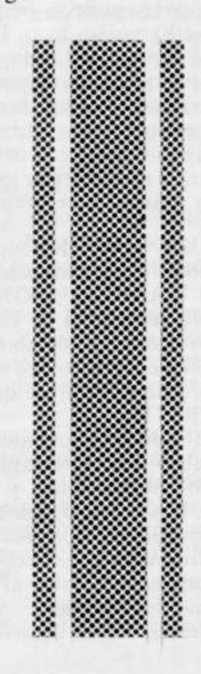
Estos dos programas se incluyen como ejemplos de programación que pueden ser usados como una guía general para que el usuario escriba sus propios programas. También demuestran la técnica de comprobar el canal de errores después de realizar cada acceso a la unidad de discos.

Dado su tamaño, tanto el Sequential File como el Random File necesitan el cartucho de ampliación de 3K para cargar correctamente en el VIC-20.

El programa DOS 5.1 no se carga directamente, sino que lo carga el programa "C-64 Wedge".

0	"1541test/demo"	zx 2a
13	"how to use"	prg
5	"how part two"	prg
4	"vic-20 wedge"	prg
1	"c-64 wedge"	prg
4	"dos 5.1"	prg
11	"copy/all"	prg
9	"printer test"	prg
4	"disk addr change"	prg
4	"dir"	prg
6	"view bam"	prg
4	"check disk"	prg
14	"display t&s"	prg
9	"performance test"	prg
5	"sequential file"	prg
13	"random file"	prg
558	"blocks free	

Fig. 1. Representación en pantalla del directorio del disco Test/Demo.



Aventuras en el Desierto



Tendrás que luchar contra las adversidades del desierto y huir de unos de unos perseguidores tenaces para poder sobrevivir el recorrido. Si no lo logras, sólo servirás de postre para los buitres.

En "Aventuras en el Desierto", un juego de habilidad para el VIC-20 no ampliado, tu conduces un jeep a través de un terreno lleno de obstáculos, intentando escaparte de unos perseguidores despiadados que viajan en un helicóptero provisto de unas bombas rastreadoras, atraídas por el calor.

Si tú puedes desviar el tubo de escape de tu jeep para que apunte hacia arriba, confundirás las capacidades rastreadoras de las bombas. Si eres capaz de evitar las zanjas y los cáctus gigantes, a lo mejor consigues llegar a tu destino.

La barra de espaciado desplaza el jeep hacia arriba, para que pueda saltar encima de las zanjas y los cáctus. La tecla F7 desvía el tubo de escape para que confunda a las bombas rastreadoras que son lanzadas con cuidado desde el helicóptero.

La letra que aparece a pie de pantalla indica la distancia recorrida.

El recorrido tiene cuatro secciones, de la A a la D. El recorrido presentado en el listado es medianamente difícil, pero yo te enseñaré a escribir un programa más difícil, por si éste te resulta demasiado fácil. (O a lo mejor quieres que sea más fácil para tu hermana la pequeña, claro).

Este programa demuestra una de las características más potentes del Basic-Microsoft la capacidad que tiene para crear gráficos animados. Como programador principiante, pronto descubrirás que cuando los gráficos animados se introducen en pantalla mediante un "Poke" la acción para.

VIC-20
NO AMPLIADO

En este momento, hay muchas personas que renuncian a seguir trabajando con el Basic y empiezan con el Forth o el Ensamblador Code. No reconocen que el imprimir los "strings" en Basic se aproxima a ser un movimiento de memoria en lenguaje máquina. El intérprete igualmente tiene que seguir la pista de muchas cosas y realizar muchas tareas, pero resulta increíblemente rápido.

Para comprobar lo rápido que resulta esta técnica, prueba el siguiente programa y consulta el manual si no entiendes cómo funciona.

```
10 A$="(22 espacios)B(22 espacios)"
20 FORI=1T023
30 PRINT CHR$(19) MID$(A$,I,22)
40 NEXTI
50 GOTO20
```

Ahora añade este retraso de tiempo para que puedas ver la B pasar por la pantalla:

```
35 FORT=1TO100:NEXT
```

Podrías decir que la B iría igual de rápido mediante el uso de los "Pokes", pero recuerda que no estamos presentando un carácter sólo con este programa; estamos imprimiendo 22 caracteres, cada uno de los cuales puede ser diferente. Además, podemos añadir colores distintos, y funcionará igual de rápido.

Para operaciones de "Pokes" hace falta introducir estos valores tanto en la pantalla como en la memoria de color de pantalla, y si quieres entrar y salir de una modalidad de multicolor, podrás observar que, en cuanto a rapidez, el uso de los "Pokes" no puede competir con el uso de sentencias Print. Prueba este programa, y después intenta hacerlo mediante los "Pokes".

```
10 A$="(color negro)*(color rojo)*(color
   cian)*(color morado)*(color verde)*
   (color azul)*(color amarillo)*(cada
   cambio de color y símbolo debe estar
   separado por dos espacios).
15 A$=A$+A$+A$+A$
20 FORI=1TO90
25 PRINT CHR$(19) MID$(A$,I,28)
30 FORT=1TO90:NEXT
40 NEXTI
50 GOTO20
```

Esta constituye la técnica básica utilizada para mover el suelo debajo del jeep; un "string" distinto desplaza los cáctus a lo largo de las dunas. Cuando esto se combina con las capacidades gráficas multi-colores, podrás ver que las ruedas del jeep siguen los altibajos del terreno sin usar un sólo "Peek" y solamente dos "Pokes".

Las ruedas que aparecen encima de cada duna, se forman en la modalidad de multicolores del mismo color de carácter que el color de la pantalla para que no se pueda ver la parte que corresponde a la rueda. Luego, al hacer un "Poke" en la memoria de color en la posición debajo de las llantas de las ruedas a un color de carácter negro, aparecen los neumáticos negros, y siempre en el suelo.

Si sustituyes el POKE 646,9 en la línea 120 por POKE646,8, podrás ver que los neumáticos están allí, lo que pasa es que son invisibles. La combinación de estas dos técnicas permite que se realice un desplazamiento horizontal a una velocidad aceptable.

El recorrido que se hace en este programa se compone de cuatro "strings" - B$(1-4). Estos "strings" se desplazan mediante un bucle For...Next comen-

zando en la línea 110 y que se imprime en la línea 120.

Los "strings" B$ a su vez se componen de diferentes combinaciónes de A1$, A2$ y A3$. Cada "string" B$d, formado en las líneas 585 y 586, se compone de cuatro "substrings". Estos "substrings" se encuentran en las líneas 580-583. Se componen de @, As, Bs, Cs, Ds y Es.

Estos caracteres gráficos especialmente creados forman el suelo (y los neumáticos).

El signo @ constituye un bloque completo, y una "A", un espacio. Los otros están incluidos dentro de estos dos extremos. En estos "substrings" las As son las zanjas, los signos @ hacen los cáctus, y los otros causan las ondulaciones del terreno.

Mi consejo es que te inventes nuevos recorridos, cuando hayas dominado éste. Las normas para la creación de los "strings" son sencillas. A1$ debe iniciar cada B$ y A2$ debe finalizar cada B$. Los primeros 22 caracteres de A1$ deben ser los mismos que los últimos 22 caracteres en A2$. Yo prefiero que el terreno no sea demasiado rocoso, así que normalmente me paso a la siguiente letra, y además, creo que es mejor evitar las zanjas durante los primeros 22 caracteres de A1$ para que el pobre jugador no se estrelle en seguida. Pero aparte de esto, se pueden crear recorridos fáciles o difíciles. Simplemente se cambia el orden de los 63 caracteres del "string".

El programa está compuesto de dos partes y se deben de salvar en cinta, una detrás de otra. El primer programa reserva una sección de memoria para caracteres especiales, y a continuación, introduce los datos en la memoria, mediante un "Poke" para la creación de los caracteres especiales. El uso de los caracteres especiales proporciona a este programa un aspecto de gráficos de alta resolución.

El programa utiliza unos caracteres de doble altura, de unos ocho pixels de anchura y unos 16 pixels de profundidad. Por este motivo los datos se ordenan por grupos de dieciséis. Si te apetece modificar cualquiera de los caracteres especiales —por ejemplo, cambiar el jeep por otro tipo de vehículo— no te debe resultar difícil. La primera sentencia Data crea el signo @, la segunda la A, la tercera la B, etc. El resto del programa saca en pantalla unas instrucciones sencillas y carga la segunda parte automáticamente.

Al teclear la segunda parte del programa, debes introducir la línea una como GOTO 510. De esta forma el programa no entrará en la modalidad de caracteres de doble altura aparte de pasar a caracteres personalizados. El programa se puede parecer una guerra con el abecedario, pero te permitirá descubrir tus errores, dado que los mensajes de error salen en inglés como piezas sueltas del jeep. Cuando parece que todo funciona, entra GOTO 500. Omití las sentencias REM dado que el VIC no ampliado no dispone de suficiente memoria, pero te ayudará la Tabla 1.

Tabla 1. Explicaciones de las variables utilizadas en el programa "Aventuras en el Desierto"

V	**- la posición de la bomba.**
CA	**- la posición en el "string" de los cáctus.**
D	**- la variable de control para el jeep saltando.**
C	**- una variable de control que visualizará, mediante un "Poke", la bomba.**
WC	**- el color del neumático, blanco o negro.**
SN	**- posición del generador de sonido.**
SV	**- posición del control de volumen (también color auxiliar).**
D$	**- lleva el cursor a "Home" y baja al nivel de la carretera.**
S$	**- una línea de bloques sólidos.**
F$	**- lleva el cursor a "Home" y va a la posición normal del jeep.**
CA$	**- una serie del cursor a la derecha, un cáctus y a continuación, más cursores a la drecha.**

```
501 POKE52,28:POKE56,28:CLR
510 FORQ=7168TO7168+26*16-1:READA:PO
KEQ,A:NEXT
515 PRINT"[CLR][3CRSRD][6SPC]CONTROL
ES":PRINT"[2CRSRD][SPC][RVSON]ESPACI
O[RVSOFF][2SPC]HACE[SPC]SALTAR[3SPC]
[CRSRD]TU[SPC]JEEP[SPC]SOBRE[SPC]LAS
"
516 PRINT"[CRSRD][SPC]ZANJAS[SPC]Y[SPC]
LOS[SPC]CACTUS."
517 PRINT"[2CRSRD][SPC][RVSON]F7[RVSOFF]
[SPC]GENERA[SPC]CALOR[SPC]PARA"
518 PRINT
519 PRINT"[SPC]CONFUNDIR[SPC]LAS[SPC]
BOMBAS","[CRSRD][4SPC]RASTREADORAS."
:POKE631,131:POKE198,1
520 DATA85,85,85,85,85,85,85,85,85,8
5,85,85,85,85,85,85
521 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0
522 DATA0,32,32,136,136,136,32,32,85
,85,85,85,85,85,85,85
523 DATA32,136,136,136,32,32,85,85,8
5,85,85,85,85,85,85,85
524 DATA136,136,32,32,85,85,85,85,85
,85,85,85,85,85,85,85
525 DATA32,32,85,85,85,85,85,85,85,8
5,85,85,85,85,85,85
526 DATA0,0,0,0,0,0,7,4,15,15,15,1
4,30,0,0
527 DATA0,0,0,0,0,0,254,146,255,25
5,135,51,121,252,252
528 DATA0,240,136,132,130,194,205,25
5,255,255,255,255,254,254,0,0
```

```
529 DATA0,0,0,0,0,192,254,254,255,
135,123,253,253,252,252
530 DATA0,0,0,0,16,16,16,80,84,84,
92,112,16,24,126
531 DATA0,48,120,204,204,204,120,48,
0,0,0,0,0,0,0,0
532 DATA254,4,4,126,147,147,255,255,
127,62,16,0,0,0,0,0
533 DATA16,0,0,0,0,130,197,242,156,0
,0,0,0,0,0,0
534 DATA143,4,4,126,147,147,255,255,
127,62,16,0,0,0,0,0
535 DATA240,0,0,0,0,133,192,245,156,
0,0,0,9,0,0,0
536 DATA0,128,160,160,192,176,204,24
2,253,119,203,203,123,49,0,0
537 DATA0,0,4,10,17,33,113,253,254,2
55,255,255,255,254,252,60
538 DATA0,0,0,0,0,0,64,234,255,251
,255,59,31,202,47
539 DATA0,0,60,195,0,0,60,195,0,0,60
,195,0,0,60,195
540 DATA4,22,30,31,63,63,63,63,31,31
,30,30,14,14,14,4
541 DATA68,84,40,16,56,56,56,56,16,0
,0,0,0,0,0,0
542 DATA65,65,0,0,20,20,20,0,0,20,20
,20,20,20,20,20
543 DATA1,0,16,20,20,20,16,1,1,16,20
,20,20,16,0,1
544 DATA65,0,4,20,21,21,21,21,21,21,
21,21,20,4,0,65
545 DATA1,0,16,20,20,20,20,20,20,20,
20,20,20,16,0,1
```

Rutinas

Las sentencias bucle del juego principal forman la primera parte del programa.

Líneas 85-87: una subrutina que tira la bomba y comprueba si ha dado en el blanco o no.

Líneas 200-250: el fin del juego.

Líneas 250-300: la rutina de chocar contra un cáctus.

Líneas 300-400: la rutina de choque.

Líneas 480-481: los datos que mueven el helicóptero.

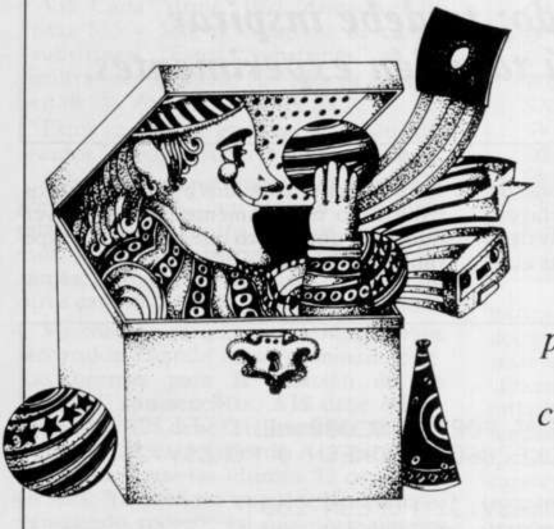
Líneas 500 etc.: simplemente inicialización.

El programa está lleno de trucos que yo aprendí después de pasar muchas horas de programación con programas de archivos, bases de datos y procesadores de textos, así que puede contener algunas cosas nuevas para los aficionados a inventar juegos. Este juego no solamente resultará divertido, sino que espero que te inspire a experimentar también.

Este programa no es solamente divertido; te debe inspirar para que tú también experimentes.

```
1 GOTO500
85 POKEV-22,19:POKEV+K-22,10:POKEV,2
1:POKEV+K,0:V=V+22:POKESN,200:C=1
86 IFV=7904THENPRINTF$;"[CRSRU]AAAA"
:GOTO300
87 RETURN
110 FORQ=1TO4:POKE7924,21+Q
115 FORW=1TO230
120 POKE646,9:PRINTD$;MID$(B$(Q),W,2
2)
122 POKE38662,WC:POKE38662,WC
125 IFCA<26THENPRINTD$;"[CRSRU]";MID
$(CA$,CA,23):GOTO130
127 IFPEEK(7920)=0THENCA=1
130 PRINT"[HOM][RED]"SPC(X+22)"ANOA"
:POKESN,200
160 IFDTHENPRINTF$;"[CRSRU][BLU]FGHI
":PRINT"[BLK]AAKAK":WC=9:D=D-1:B=1:G
OTO180
165 IFBTHENPRINTF$;"[CRSRU]AAAA":WC=
8:B=0
170 PRINTF$;"[BLU]FGHI":IFPEEK(197)=
32THEND=6
175 IFPEEK(7904)=1GOTO300
177 IFPEEK(7883)=10THENGOSUB250
180 READX:IFX=77THENPRINTCHR$(19)SPC
(20)"AAA":RESTORE:GOTO180
182 PRINT"[HOM][RED]"SPC(X+22)"ALMAA
"
185 IFX=3THENIFPEEK(36878)=26THENGOS
UB85:GOTO189
188 POKESN,0:IFCTHENPOKESV,26:POKEV-
22,19:POKEV+K-22,10:V=7750:C=0
189 IFPEEK(197)=63THENIFD=0THENPOKES
V,42:C=1:IFX=3THENREADX:GOTO189
190 CA=CA+1
199 NEXT:NEXT
200 PRINT"[CLR]":POKE36869,240:POKE3
6867,PEEK(36867)AND254
210 PRINT"[4CRSRD][3CRSRR]FELICIDADE
S":END
250 FORI=16TO31STEP.3:POKESV,I:POKES
N,249
254 FORI=16TO31STEP.3:POKESV,I:POKES
N,249
260 IFI>24THENPOKE7860,20:POKE7860+K
,0
280 FORT=1TO40:NEXT
285 NEXT:FORT=1TO2000:NEXT
295 POKE7860,1:POKESN,0:POKESV,26:RE
TURN
300 POKESV,31:POKESN,200
305 POKESV,31:POKESN,200
310 PRINTF$;"[BLU]AFGHI"
311 POKE38622,9
320 FORT=1TO200:NEXT
330 FORS=31TO16STEP-1
332 POKESV,S:POKE36876,0:POKE36877,1
35
340 PRINTF$;"AAAAAA[CRSRD][3CRSRL][BLU]
APQR[CRSRU][CRSRL][BLK]T"
360 FORT=1TO100:NEXT
370 NEXT:POKE36877,0:RUN
480 DATA19,19,18,18,17,17,16,16,15,1
5,14,14,13,13,12,12,11 10,10
481 DATA9,9,8,8,7,7,6,6,5,5,4,4,3,3,
3,3,3,3,3,3,3,2,1,0,-1,+2,77
500 POKE36869,255:POKE36867,PEEK(368
67)OR1:POKE36879,31
510 K=30720
580 A1$="CCDEDCBCDEDCBDE@EDCDE@EDBDD
CBCDEDAAAADCBCBCDEDCDEDE@ECBCBCCBCDC
"
582 A2$="CCDEDCBCDEDCBDE@EDCDCBAAAAD
CBCDEDEDEDE@EDDCDEDCBCDEDCBDE@EDCDED
"
583 A3$="CCAAAABCDE@EDDE@EDCDCBAAAAD
CBCDEDEDEDAAAABCDEDBDE@ECECBCBCCBCDC
"
585 B$(1)=A1$+A2$+A1$+A2$:B$(2)=A1$+
A2$+A1$+A2$
586 B$(3)=A1$+A2$+A3$+A2$:B$(4)=A1$+
A3$+A2$+A3$
590 D$="[HOM][10CRSRD]":F$=LEFT$(D$,
10)+"[CRSRR]"
592 CA$="[19CRSRR][GRN]AJAA[22CRSRR]
"
594 S$="@@@@@@@@@@@@@@@@@@@@@@"
595 POKE646,10:PRINT"[CLR]";D$;S$;S$
;S$:WC=8:CA=26:X=19:V=7728:SV=36878:
SN=36877
597 PRINT"[HOM]":FORI=1TO8:PRINT"AAA
AS":NEXT:POKESV,26
598 PRINTF$;"[RED]FGHI"
599 GOTO110
```

El Bombardero Loco



Una ciudad está siendo atacada por unas bombas en forma de letra; los habitantes temen por su vida. ¿Dónde está Superman cuando lo necesitamos? Solamente tu agilidad mental y lo bien que conoces el teclado pueden salvar la ciudad.

Qué es más importante: aprender con tu ordenador personal o divertirte con él? Seguramente eso depende de que si eres un niño o el padre del niño. Pero "El Bombardero Loco" resulta tan divertido como educativo.

Con este juego puedes aprender a escribir a máquina, o por lo menos familiarizarte con el teclado. Y como cualquier persona metida en el mundo de los negocios sabe, los teclados están por todos lados.

El "Bombardero Loco" ataca Cualquier Ciudad, El Mundo, con unas bombas realmente malévolas, en forma de letras. Puede atacar muy de prisa o muy despacio, como tú quieras. La facilidad con que tú manejas el teclado debe fijar la velocidad.

Una sirena rompe estrepitosamente la tranquilidad de la pantalla del título y las bombas (de letras) empiezan a atacar. El sonido que lleva el programa es realmente bueno, y cuanto mejor el altavoz del monitor o de la televisión, mejor resultará el sonido.

Te piden que elijas una velocidad, y empieza la diversión. Una ciudad cubierta de niebla sale en pantalla. De repente, la pantalla te avisa mediante una alarma destellante de "Alerta Roja".

"El malo" se asoma entre las nubes. Pero no le hagas mucho caso solamente está intentando distraerte para que la bomba lanzada al azar pueda llegar a tierra.

C-64

A veces la bomba (en forma de letra) sale por debajo del "Bombardero Loco", pero normalmente cae desde otro lugar del cielo. Tú tienes que reconocer la letra y pulsar la misma letra en el teclado antes de que la bomba gire y explote sobre la ciudad.

Si tú eres rápido y aciertas, la bomba se destruye a sí misma, caerá y explotará. Entonces, el "Bombardero Loco" se vuelve a esconder entre las nubes.

Pero no te vayas. El volverá para descargar un total de 21 bombas o las que necesite para ganar.

Si consigues responder con las mismas letras, tú ganarás, y Cualquier Ciudad te lo agradecerá con un fuerte aplauso. A continuación, el programa te da la oportunidad de elegir una velocidad y volver a empezar.

Pero si fracasas, la ciudad explota, te indican el número de bombas que conseguiste parar, y te dan la oportunidad de jugar de nuevo.

A propósito

Ten en cuenta también que una niebla mañanera aparece durante la primera parte del ataque. Después de haber atacado diez veces, el "Bombardero Loco" te deja descansar un poco. Cuando el ataque vuelve a comenzar, la niebla desaparece, pero una nube cada vez más espesa dificulta el reconocimiento, y por lo tanto, la destrucción, de las bombas. Después de cada ataque, la niebla o las nubes se van espesando y subiendo cada vez más.

Tabla. Explicación línea por línea del programa "Bombardero Loco".

- 3 Título y sirena.
- 4 Inicialización del sprite. BX y BY son las variables para la posición del sprite.
- 5 Inicializa las variables de sonido y fija el color del "Bombardero Loco" en amarillo mediante un "Poke".
- 6 Solicita la velocidad y posiciona el "Bombardero Loco" en la esquina superior izquierda de la pantalla mediante un "Poke".
- 7-15 Introduce la velocidad de las bombas.
- 16 MC es la variable que cuenta las bombas.
- 20-22 Convertir la pantalla en negro mediante un "Poke" y verificar que ha transcurrido la mitad del juego -- MC $\geq$ 20.
- 23-25 Esta es la subrutina que indica la mitad del juego. Si MC $\geq$ 20, el programa la ejecuta.
- 30 TT sigue la pista de la bomba en la pantalla.
- 31-36 La "Alerta Roja" y la sirena.
- 37 Comienza la doble verificación de la entrada en el bucle principal.
- 40 N seleccionará dónde aparecerán las bombas en la pantalla (tab=N).
- 50 La pantalla se vuelve negra mediante un "Poke" y desplaza el cursor a la posición "Home" (CHR$(19)).
- 70-82 Define la ciudad y el cielo.
- 94 CR prohíbe que la niebla y las nubes sobrepasen los límites de la pantalla.
- 95-98 Mediante unos "Pokes" al azar, la niebla o las nubes salen en pantalla.
- 100 LE selecciona la letra que será lanzada como bomba.

110 Asegura que la bomba es una letra y no un número ni un símbolo gráfico.
120 GETK$ empieza a buscar una respuesta del teclado para la bomba en forma de letra.
130 Si la respuesta es correcta, esta subrutina llama la explosión en la rutina 155-250.
155-250 La letra se cae, explota, y el Bombardero Loco se esconde.
170-178 La bomba parece explotar.
200 El sonido se apaga mediante un "Poke".
250 Incrementa el contador MC. Hace desaparecer el sprite mediante un "Poke" y vuelve a recoger otra bomba.
300-350 Sonido de la bomba cayendo.
310 Convierte la N (valor del tab) en el valor para el ángulo del giro de la bomba lanzada.
311-312 Mantienen la bomba dentro de los límites de la pantalla.
360 Detecta el momento en que la bomba llega a la ciudad.
370 Borra las letras y deja una huella roja donde éstas se encontraban.
800-808 La pantalla y la ciudad parecen explotar.
810-820 Ultima pantalla.
820 Vuelve a sonar la sirena.
830-854 Pregunta al jugador si quiere seguir o no.
1050-1054 Sonido de la explosión.
2000-2028 Pantalla del título.
2029 Lee las líneas Data del sprite e introduce a éste en la correcta posición de memoria mediante un "Poke".
2030-2080 Sonido de la sirena.
4000-4070 Rutina del ganador.
6000-6003 Líneas Data que definen el sprite del Bombardero Loco.

```
1 REM MAD L BOMBER
3 GOSUB2005
4 V=53248:POKEV+21,0:POKE2042,13:BX=
80:BY=30
5 SV=54296:SW=54276:SA=54277:SH=5427
3:SL=54272:JUMP=0:POKEV+41,7
6 PRINT"VELOCIDAD[SPC]DEL[SPC]JUEGO"
:POKEV+4,BX:POKEV+5,BY
7 PRINT:PRINT"1-SUPER[SPC]RAPIDO":PR
INT"2-RAPIDO":PRINT"3-MEDIO":PRINT"4
-LENTO"
8 INPUT"5-SUPER[SPC]LENTO[6SPC][6CRSRU]
";XP
9 IFXP=1THENSP=0
10 IFXP=2THENSP=75
11 IFXP=3THENSP=150
12 IFXP=4THENSP=300
13 IFXP=5THENSP=600
14 IFXP>5THENPRINTCHR$(147):GOTO6
15 IFXP<1THENPRINTCHR$(147):GOTO6
16 MC=0:PRINTCHR$(147)
20 IFMC>20THEN4010
21 IFMC<>10GOTO30
22 IFMC=10THENFORX=1TO4:POKE53281,1:
POKE53280,1:POKE53291,0:POKE53280,0:
NEXTX
23 PRINTCHR$(147):FORX=1TO7:PRINT:NE
XTX
24 PRINTCHR$(18)"[SPC]MUY[SPC]BIEN-E
STAS[SPC]EN[SPC]LA[SPC]MITAD[SPC]DEL
[SPC]JUEGO":PRINTCHR$(19):JUMP=0
25 GOSUB2031
30 TT=1:X=0
31 FORLP=0TO2
32 POKESV,15:POKESW,33:POKESA,0:POKE
SH,45:POKESL,0
33 PRINTCHR$(19)"ALERTA[SPC]ROJA-EL[SPC]
BOMBARDERO[SPC]LOCO[SPC]SE[SPC]ACERC
A":FORLO=1TO100:NEXTLO
34 PRINTCHR$(19)CHR$(18)"ALERTA[SPC]
ROJA-EL[SPC]BOMBARDERO[SPC]LOCO[SPC]
SE[SPC]ACERCA"
35 POKESV,15:POKESW,33:POKESA,0:POKE
SH,0:POKESL,0:POKEV+21,4
36 FORLO=1TO40:NEXTLO:NEXTLP
37 IFK$=CHR$(X)THEN40
40 N=INT(RND(1)*38)+1
41 IFJUMP>0THENGOTO82
42 JUMP=1
50 POKE53281,0:POKE53280,0:PRINTCHR$
(5);CHR$(19)
70 FORX=0TO39:POKE1944+X,120:POKE562
16+X,7:NEXTX
75 FORX=0TO3:LD=INT(RND(1)*30)+1:POK
E1904+LD,102:POKE56176+LD,13
77 POKE1864+LD,92:POKE56136+LD,13:NE
XTX
80 FORX=0TO5:LD=INT(RND(1)*35)+1:POK
E1904+LD,102:POKE56176+LD,1:NEXTX
82 FORX=0TO39:POKE1024+X,42:POKE5529
6+X,12:NEXTX
94 CR=CR-40:IFCR<-800THENCR=-800
95 FORX=0TO24:LD=INT(RND(1)*35)+1
96 POKE1864+LD+CR,42:POKE56136+LD+CR
,1:NEXTX
97 FORX=0TO4:LD=INT(RND(1)*35)+1
98 POKE1144+LD,42:POKE55416+LD,1:NEX
TX
100 LE=INT(RND(64)*90)+1
110 IFLE<65ORLE>90GOTO100
120 GETK$:IFK$=""THEN302
130 IFK$=CHR$(LE)THEN155
140 GOTO302
155 POKESV,15:POKESW,133:POKESA,0:PO
KESH,0:POKESL,130
156 FORFRY=1TO2
157 BX=BX+4:BY=BY-8:POKEV+4,BX:POKEV
+5,BY
170 PRINTCHR$(13)CHR$(145)CHR$(145)T
AB(N);:PRINT"/"
172 PRINTCHR$(13)CHR$(145)CHR$(145)T
AB(N);:PRINTCHR$(LE):BX=BX+4:BY=BY-8

173 PRINTCHR$(13)CHR$(145)CHR$(145)T
AB(N);:PRINT"-":POKEV+4,BX:POKEV+5,B
Y
174 PRINTCHR$(13)CHR$(145)CHR$(145)T
AB(N);:PRINT"*":BX=BX+4:BY=BY-8
175 PRINTCHR$(13)CHR$(145)CHR$(145)T
AB(N);:PRINTCHR$(28)".":POKEV+4,BX
176 PRINTCHR$(13)CHR$(145)CHR$(145)T
AB(N);:PRINTCHR$(LE):POKEV+5,BY
178 PRINTCHR$(13)CHR$(145)CHR$(145)T
AB(N);:PRINT"+":PRINTCHR$(5)
179 NEXTFRY
180 POKESV,15:POKESW,129:POKESA,15:P
OKESH,40:POKESL,200
182 POKEV+21,0:FORPL=1TO100:NEXTPL
185 POKESV,0:POKESW,0:POKESA,0:POKES
H,0:POKESL,0
187 POKESV,15:POKESW,129:POKESA,15:P
OKESH,40:POKESL,200
188 POKESV,9:FORPL=1TO20:NEXTPL:POKE
SV,6:FORPL=1TO20:NEXTPL:POKESV,0
189 POKESW,0:POKESA,0:POKESH,0:POKES
L,0
190 POKESV,15:POKESW,129:POKESA,15:P
OKESH,40:POKESL,200
```

```
191 FORPL=1TO350:NEXTPL
195 POKEV+21,0:POKEV+5,24:POKEV+4,60

200 POKESV,0:POKESW,0:POKESA,0:POKES
H,0:POKESL,0
250 MC=MC+1:PRINTCHR$(19):BX=60:BY=6
0:POKEV+21,0:GOTO20
302 POKESV,15:POKESW,13:POKESA,240:P
OKESH,44:POKESL,0
304 POKESV,0:POKESW,0:POKESA,0:POKES
H,0:POKESL,0
306 TW=INT(RND(1)*2)
308 IFTW=0THENBD=1
309 IFTW=1THENBD=-1
310 N=N+BD
311 IFN<3THENN=3
312 IFN>37THENN=37
325 PRINTCHR$(13)CHR$(145)TAB(N);:PR
INTCHR$(LE)
350 FORL=1TOSP:NEXT
360 TT=TT+1:IFTT=12THEN800
370 PRINTCHR$(145)TAB(N)CHR$(28)CHR$
(46):PRINTCHR$(5)
372 BX=BX+8:BY=BY+8:POKEV+4,BX:POKEV
+5,BY
380 GOTO120
800 FORBD=1TO5:POKE53281,1:POKE53280
,1:POKE53281,0:POKE53280,0:NEXT
802 POKEV+21,0:POKEV+4,24:POKEV+5,60

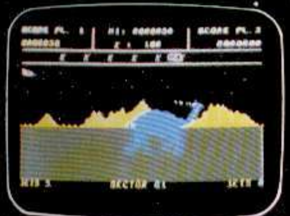
805 PRINTCHR$(147):FORX=1TO5:PRINT:N
EXT:PRINTTAB(13)"OH[SPC]QUERIDO!!!":
GOSUB1050
808 PRINTCHR$(147):FORX=1TO5:PRINT:N
EXT
810 PRINTTAB(8)"TIENES[SPC]QUE[SPC]S
ER[SPC]MAS[SPC]RAPIDO":PRINT
811 PRINTTAB(9)"PARA[SPC]SALVAR[SPC]
LA[SPC]CIUDAD[SPC]DEL"
812 PRINT:PRINTTAB(7)"****[SPC]BOMBA
RDERO[SPC]LOCO[SPC]****"
814 PRINT:PRINTTAB(5)"HAS[SPC]DESTRU
IDO";MC;"BOMBAS-LETRA[SPC]!":PRINTCH
R$(19)
815 FORX=1TO1000:NEXTX
820 GOSUB2031
830 PRINTCHR$(147):FORX=1TO5:PRINT:N
EXT:PRINTTAB(5)"QUIERES[SPC]JUGAR[SPC]
OTRA[SPC]VEZ[SPC](S/N)?"
840 GETA$:IFA$=""THEN840
850 IFA$="S"THENPRINTCHR$(19):GOSUB2
031:CLR:GOTO4
852 IFA$="N"THEN860
854 IFA$<>"N"THEN830
860 END
1050 FORXX=35TO0STEP-.2:POKESV,XX:PO
KESW,129:POKESA,15:POKESH,40:POKESL,
200:NEXT
1052 FORXX=35TO0STEP-.2:POKESV,XX:PO
KESW,129:POKESA,15:POKESH,40:POKESL,
200:NEXT
1054 POKESV,0:POKESW,0:POKESA,0:POKE
SH,0:POKESL,0:RETURN
2005 SV=54296:SW=54276:SA=54277:SH=5
4273:SL=54272:V=53248
2010 PRINTCHR$(147):POKE53281,0:POKE
53280,0:PRINTCHR$(5)
2020 FORPT=1TO6:PRINT:NEXT:PRINTTAB(
11)"EL[SPC]BOMBARDERO[SPC]LOCO"
2022 PRINT:PRINTTAB(13)"BY[SPC]GARY[SPC]
FIELDS"
2023 PRINT:PRINTTAB(18)"1983"
2024 FORX=1TO400:NEXTX
2025 PRINT:PRINT:PRINT"TU[SPC]MISION
[SPC]ES[SPC]PROTEGER[SPC]LA[SPC]CIUD
AD[SPC]DE[SPC]LAS"
2026 PRINTTAB(11)CHR$(18)"[SPC]BOMBA
S-LETRA!![SPC]"
2027 PRINT"DESTRUYE[SPC]LAS[SPC]BOMB
AS-LETRA[2SPC]QUE[SPC]CAIGAN"
2028 PRINT"PULSANDO[SPC]LA[SPC]MISMA
[SPC]LETRA[SPC]EN[SPC]EL[SPC]TECLADO
":FORX=1TO1000:NEXTX
2029 PRINTCHR$(19):FORPR=0TO63:READP
Q:POKE832+PR,PQ:NEXTPR
2031 FORDL=0TO500:NEXTDL
2035 FORL=0TO24:POKESL+L,0:NEXT
2040 POKESL+14,5:POKESL+18,16:POKESL
+3,1:POKESL+24,143:POKESL+6,240:POKE
SL+4,65
2050 FR=5389
2060 FORFT=1TO50:FQ=FR+PEEK(SL+27)*3
.5
2070 HF=INT(FQ/256):LF=FQ-HF*256:POK
ESL+0,LF:POKESL+1,HF
2075 PRINTTAB(10)"VAN[SPC]A[SPC]CAER
[SPC]LAS[SPC]BOMBAS":NEXT:POKESL+24,
0
2080 PRINTCHR$(147):RETURN
4010 PRINTCHR$(147):FORX=1TO5:PRINT:
NEXT:PRINTTAB(10)"TIENES[SPC]DEDOS[SPC]
MUY[SPC]HABILES"
4012 PRINTTAB(16)"-TU[SPC]GANAS-":PR
INT:PRINT
4014 PRINTTAB(7)"ESTALLASTE";MC;"BOM
BAS-LETRA"
4015 PRINT:PRINT"LA[SPC]GENTE[SPC]CO
NTENTA[SPC]TE[SPC]LO[SPC]AGRADECE[SPC]
!"
4040 FORL=0TO24:POKESL+L,0:NEXT:POKE
SL+0,240:POKESL+1,33:POKESL+5,8
4050 POKESL+22,104:POKESL+23,1:POKES
L+24,79:FORN=1TO19:POKESL+4,129
4060 FORXT=1TO100:NEXT:POKESL+4,128:
:FORXT=1TO30:NEXT:NEXT
4065 POKESL+22,0:POKESL+33,0:POKESL+
24,0:POKESL+4,0
4067 POKESL+1,0:POKESL+0,0:POKESL+1,
0:POKESL+5,0:POKESL+22,0:POKESL+23,0

4068 POKESL+33,0
4069 FORX=0TO999:NEXTX
4070 GOTO830
6000 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0
6001 DATA0,0,0,0,0,0,12,0,128,30,64,
199,152,32,224,218
6002 DATA32,176,108,38,184,127,33,14
3,255,248,255,254,29,254,0,28
6003 DATA7,255,157,3,255,248,1,255,1
93,0,0,6,0,0,0,0
```

indescomp
SOFTWARE

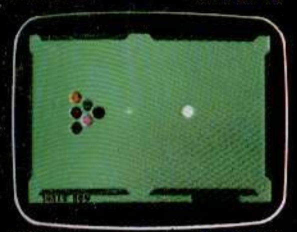
UNA NUEVA GENERACION DE VIDEOJUEGOS

COMMODORE 64


MUTANT CAMELS Cass. 2.200


MATRIX Cass. 2.000


CHOPFLITER Cart. 8.170

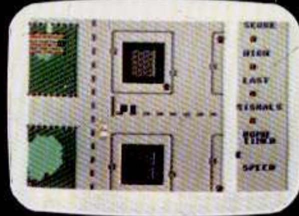

HUSTLER Cass. 2.200


REVENGE MUTANT CAMELS Cas. 2.200


CYCLONS Cass. 2.200


HOVER BOVER Cass. 2.200

COMMODORE 64


SIREN CITY Cass. 1.900


CUDDLY CUBURT Cass. 1.900


KONG Cass. 2.000


SKRAMBLE Cass. 1.700


MOON BUGGY Cass. 2.000

COMMODORE 64

• SPRITE MAN	1.900
• FROGGER 64	1.700
• SNAKE	1.800
• ANIHILATOR	1.600
• PAKACUDA	1.700
• EXTERMINATOR	1.900
• VORTEX RAIDER	1.700
• DEFENDER 64	1.700
• LASER ZONE	1.700
• CHOPFLITER	8.170
• ZAXXON	7.190
• JUMPMAN	7.190
• FORT APOCALIPSE	6.290
• FUTBOLIN	1.800
• HORACIO	2.200
• PROCESADOR TEXTOS	3.300
• MAILING Y ETIQUETAS	3.300
• CONTABILIDAD PERSONAL	3.300
• BASE DE DATOS FICHERO	3.300
• ULTRABASIC	2.900
• SINTHY 64	2.900
• FORTH TINY	2.900

SINCLAIR SPECTRUM*

- PENETRATOR
- TIME GATE
- EL JUGADOR DE AJEDREZ
- ASTROBLASTER
- COMECOCOS
- HORMIGAS
- TRAXX
- DISEÑADOR DE JUEGOS
- LA PULGA (BUGABOO)
- BILLAR AMERICANO
- EL PINTOR
- MISILES
- RAPTORES DE LA GALAXIA
- WINGED AVENGER
- INVASORES & GALAXIANS
- FORTH
- BASE DE DATOS
- MATEMATICAS
- ARITMETICAS BASICAS
- TUTOR

APPLE
900 PROG.

ATARI
800 PROG.

SINCLAIR ZX81
100 PROGRAMAS

DRAGON 32
60 PROGRAMAS

VIC 20
200 PROG.

IBM
350 PROG.

SOLICITE GRATIS CATALOGO GENERAL INDICANDO EL ORDENADOR SOBRE EL QUE LO DESEA

Ya a la venta en


El Corte Inglés


Sonitel

CHIPS & TIPS

Distribuidores autorizados y por Correo


BUGABOO LA PULGA Cassette 2.000

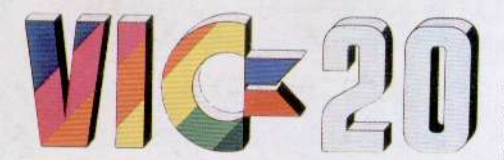

ENVIAR A: indescomp, Castellana, 179. Tel. 6563012. Madrid-16

CANTIDAD	PRODUCTO	PRECIO UNIT.	TOTAL
EMPAQUETADO Y TRANSPORTE	GASTOS DE ENVIO	300	
Incluyo cheque nominativo a favor de INDESCOPM, S.A. por .... Pts.		TOTAL	

Remitan el pedido contra reembolso a:
D. .... Dirección ....
n.º .... Provincia .... Tel. .... Profesión ....

* Los programas para Spectrum son comercializados únicamente y en exclusiva por Investronica, S. A.

Estamos en la Feria de la Informática.
"EL CORTE INGLES" Castellana, del 4 al 28 de Abril.



Microprocesador: 6502 de MOS TECHNOLOGY de 8 bits.
Memoria: 5 Kbytes de RAM ampliables a 32 K 20 Kbytes de ROM ampliables a 28 K.
Pantalla: 23 líneas de 22 caracteres. Modulador para conectar a un televisor normal. Salida monitor video.
Colores: 8 para el marco, 16 para el fonde de la pantalla y 8 para los caracteres individuales, video inverso.
Gráficos: Semi-gráficos por teclado y alta resolución por redefinición del generador de caracteres (situándolo en RAM). Definición de 176 por 184 puntos.
Teclado: Tipo QWERTY de 62 teclas más cuatro de función definibles por el usuario.
Sonido: Tres voces de tres octavas cada una decaladas una octava entre sí, resultando una extensión total de cinco octavas. Un generador de ruido aleatorio afinable para efectos especiales, un control general de volumen.
Programación: Lenguaje BASIC, intérprete residente en ROM de 8K. Posibilidad de interceptar las funciones del Basic para crear nuevas instrucciones "a medida". El Basic del Vic es uno de los rápidos actualmente en el mercado.
Complementos: Port de usuario de 8 bits entrada/salida más dos señales de sincronismo.
Bus de expansión para ampliaciones de memoria y periféricos.
Port de juegos con conexión para dos potenciómetros (paddles), y una palanca de juegos (joystick).
Almacenamiento de masa: Unidad de cassette C2N de diseño especial para registrar programas y datos.
Ampliación de memoria: En caso de ser necesario conectar más de un cartucho al mismo tiempo, está disponible un módulo (VIC 1020) que permite la conexión simultánea de hasta seis cartuchos.

VIC-1541 UNIDAD DE DISCO

Capacidad total: 174848 bytes por disco.
Secuencial: 168656 bytes por disco.
Entradas de directorio: 144 por disco.
Sectores por pista: De 17 a 21.
Bytes por sector: 256.
Pistas: 35.
Bloques: 683 (644 bloques libres).
Soportes de información: Discos standar de 5 1/4 pulgadas, de una sola cara y densidad simple.
Sistema operativo: DOS de COMMODORE inteligente (tiene procesador propio y no ocupa memoria del ordenador central).

VIC-1525 IMPRESORA

Método de impresión: Matriz de 5 × 7 puntos, impacto por un solo martillo.
Modo caracteres: Mayúsculas y minúsculas, símbolos, números y caracteres gráficos del VIC-20.
Modo gráfico: Puntos direccionables (bit image). Siete puntos verticales por columna, 480 columna máximo.
Velocidad: 30 caracteres/segundo, de izquierda a derecha, unidireccional.
Caracteres/Línea: Máximo 80. (Posibilidad de impresión en doble ancho).
Espaciado entre líneas: 6 líneas/pulgada —modo caracteres, 9 líneas/pulgadas— modo gráfico.
Alimentación de papel: Arrastre por tractor.
Ancho de papel: Entre 4,5 y 10 pulgadas.
Copias: Original más dos copias.

CARTUCHOS

Ayuda programador: Facilita la edición y depuración de programas en Basic. Instrucciones y comandos: RENUMBER, MERGE, FIND, CHANGE, DELETE, AUTO, TRACE, STEP, OFF, KEY, EDIT, PROG, DUMP, HELP y KILL.
Super expander: Intercepta el Basic del VIC permitiendo incrementar sus instrucciones y comandos en aplicaciones gráficas de sonido y juegos. Instrucciones y comandos: KEY, GRAPHIC, COLOR, POINT, REGION, DRAW, CIRCLE, PAINT, CHAR, SCNCLR, SOUND, RGE, RCOLR, RDOT, RPOT, RPEN, RJOY y RSND.
Monitor de lenguaje máquina: Facilita enormemente la depuración de programas en lenguaje máquina, es ideal como complemento del Basic para redactar y poner en marcha rutinas de alta velocidad y manejo de datos en tiempo real. Instrucciones y comandos: ASSEMBLE, BREAKPOINT, DISASSEMBLE, ENABLE, VIRTUAL ZERO PAGE, FILL MEMORY, GO, HUNT, INTERPRET, JUMP TO SUBROUTINE, LOAD, MEMORY, NUMBER, QUICK TRACE, REGISTERS, REMOVE BREACPOINTS, SAVE, TRANSFER, WALK y EXIT TO BASIC.
Además existen cartuchos de ampliación de memoria de 3,8 y 16 Kbytes.

CURSO DE INTRODUCCION AL BASIC PARTE I y II.

En forma de libro se ha editado la primera y segunda parte de un curso de Basic que parte "de cero" y está basado en el VIC-20. Van acompañados de dos cassettes con programas y ejercicios para autocontrol.

PLOTTER VIC 1520

Método de impresión: Dibujo mediante bolígrafos de diseño especial.
Color: 4 colores; negro, azul, verde y rojo con cambio desde programa.
Cabezal: Ploter X-Y tipo tambor.
Velocidad de impresión: Media de 14 car./seg.
Caracteres por línea: Máximo 80 carac., formatos de 80, 40, 20 y 10 carac./línea.
Juego de caracteres: 96
Velocidad de dibujo:264 pasos/seg.
Longitud del paso: 0,2 mm. en dirección X e Y.
Velocidad de dibujo de línea: 52,8 mm./seg. en dirección X e Y. 73 mm./seg. en una línea a 45 grados.
Area de dibujo: 480 pasos (96 mm.) en dirección X. Programable en dirección Y (Máx. + — 999 de una sola vez).
Papel: Rollo de 4,5 pulgadas (114 mm.).

MONITOR EN COLOR C-1701

Pantalla: 13 pulgadas (330 mm.).
Capacidad de representación: 25 líneas de 40 caracteres.
Resolución: 320 líneas horizontales.
Compatibilidad: VIC-20 y COMMODORE 64.
Conectable a un registrador de video.
Amplificador y altavoz: Incorporados.



COMPUTER



microelectrónica
y control, s.a.

Taquigrafo Serra, 7 5.° Telf. 250 51 03. BARCELONA-29
Princesa, 47 3.° G. Telf. 248 95 70. MADRID-8